8-2011

SIX DEGREE OF FREEDOM MINIATURE *IN VIVO* ROBOT FOR LAPAROENDOSCOPIC SINGLE-SITE SURGERY

Ryan L. McCormick
University of Nebraska-Lincoln, rlmccorm@gmail.com

SIX DEGREE OF FREEDOM MINIATURE *IN VIVO* ROBOT FOR LAPAROENDOSCOPIC SINGLE-SITE SURGERY

by

Ryan L. McCormick

A THESIS

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Master of Science

Major: Mechanical Engineering

Under the Supervision of Professor Shane M. Farritor

Lincoln, Nebraska

August, 2011

SIX DEGREE OF FREEDOM MINIATURE *IN VIVO* ROBOT

FOR LAPAROENDOSCOPIC SINGLE-SITE SURGERY

Ryan Lucas McCormick, M.S.

University of Nebraska, 2011

Advisor: Shane M. Farritor

The shift in surgery from open procedures to minimally invasive surgery (MIS) techniques have provided benefits of decreased recovery time, improved cosmetic results, and reduced costs. As advances in MIS move to minimize the number of external incisions, such as with Laparoendoscopic Single-Site (LESS) surgery, additional complexities are introduced. These complexities, including unintuitive controls, reduced dexterity, and limited workspace, hinder these methods from more widespread implementation in more complicated surgical procedures.

Through the use of a miniature *in vivo* robotic surgical platform designed for LESS surgery, these complexities can be mitigated, allowing for wider adoption of MIS by placing the entire robotic platform inside the peritoneal cavity. This thesis presents the design, prototyping, and implementation of a two armed surgical robot for use in LESS procedures. Each arm of the robot will be individually inserted through a single incision in or around the umbilicus before being mated together through the use of a central assembly rod. The robotic platform will provide increased dexterity, larger workspace, and more intuitive controls as compared to currently available technologies.

A remote surgical user interface will allow the surgeon to perform surgical procedures in all quadrants of the peritoneal cavity, as is often required for surgeries such

as colon resection. The feasibility of this platform has been demonstrated through multiple *in vivo* LESS procedures, including a cholecystectomy, colon resection, and small bowel resection in a live porcine model.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

Advances in minimally invasive surgery (MIS), such as laparoscopy, have provided patients with the benefits of faster recovery times, improved cosmetics, and reduced costs. Despite these benefits, the nature of inserting long instruments through multiple small incisions in the peritoneum for laparoscopic procedures results in unintuitive controls, lack of dexterity, and poor visual feedback. Because of these limitations, more complex procedures are rarely completed laparoscopically.

As technology allows MIS to transition to less invasive methods of surgery, a subset of laparoscopy known as Laparoendoscopic Single Site (LESS) surgery aims at reducing the number of incisions to one. Through LESS, all surgical instruments are inserted into the body through a single incision in or around the umbilicus, leaving "virtually scarless" cosmetic results. A transition from a standard open procedure to the use of LESS would reduce a patient's hospital stay from 4-6 days to 1-2 days [1]. Unfortunately, this method has additional complexities associated with the instruments passing through a single incision. In order to provide proper triangulation and vision for the surgeon, the instruments of LESS surgery are articulating, bent, or flexible. Because of this, the tools are crossed at the incision, resulting in collateral hand movements in which the surgeon's right hand controls the left end effector and vice versa.

This thesis presents the design and *in vivo* feasibility testing of a six degree of freedom (DOF) miniature *in vivo* surgical robot designed for LESS surgery, as shown in Figure 1. This two-armed surgical robot can be inserted into the peritoneal cavity through a single incision in the umbilicus before being assembled inside the cavity. This

robot is controlled by a surgeon at a remote user interface to perform surgical tasks in all quadrants of the peritoneal cavity, providing increased dexterity and more intuitive controls as compared to traditional LESS surgery techniques.

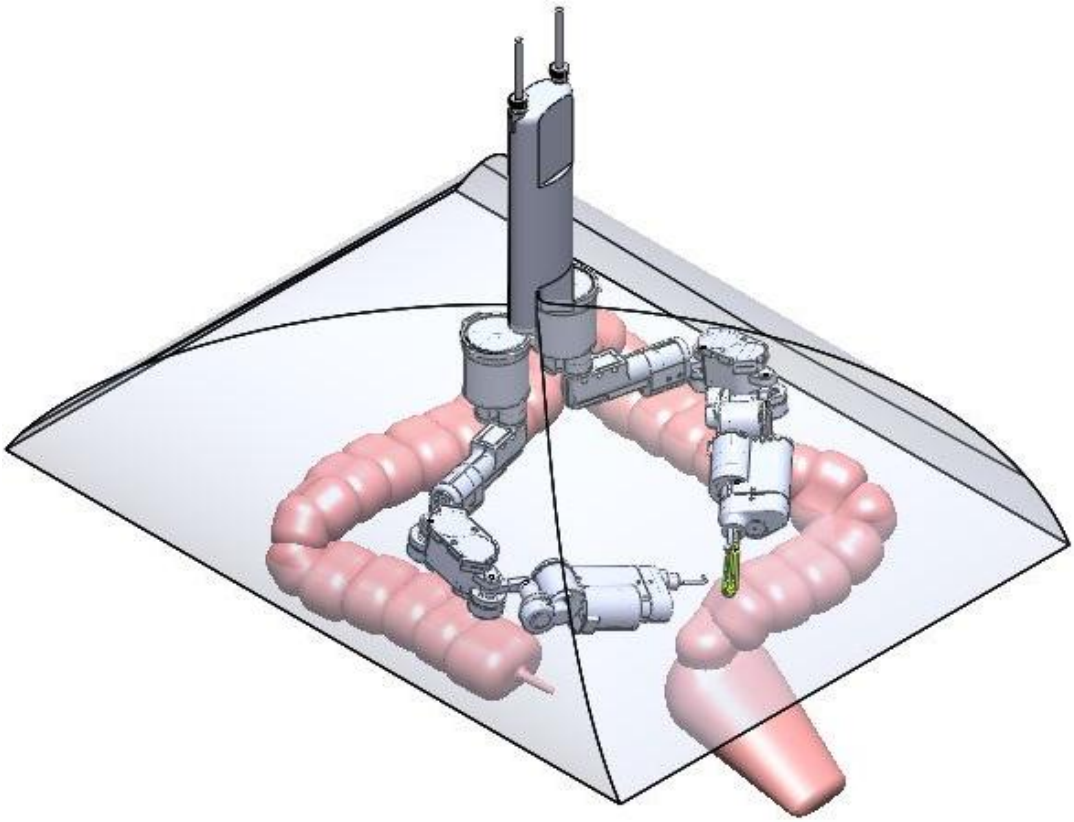

Figure 1: SolidWorks model of surgical robot inside the peritoneal cavity

Chapter 2: Background

Section 2.1: Standard Laparoscopic Surgery

During the 1990s, technological advances in MIS brought about a transition from standard open surgical procedures to laparoscopic procedures [2-3]. By replacing large incisions used in traditional open surgeries with a series of multiple 5 - 15 mm incisions to perform surgical tasks, patients have been able to enjoy quicker recovery times and reduced pain [4].

Section 2.2: Laparoendoscopic Single-Site (LESS) Surgery

A possible next step in the evolution of MIS is through the implementation of LESS. Various studies have been performed to test the feasibility of LESS in a wide range of surgical procedures. One such study, performed by Ahmed *et al.*, reviewed 102 studies of LESS procedures, including cholecystectomies, appendectomies, and nephrectomies. These procedures resulted in operative time, hospital stay, and complications comparable to conventional laparoscopy and a low rate of conversion to conventional laparoscopy [5]. In Fader and Escobar [6], LESS surgery for gynecologic oncology is examined. In this study, nine laparoscopic procedures were performed utilizing LESS. These preliminary studies have helped demonstrate the feasibility of LESS for select patients.

Section 2.3: Robots for Laparoscopic Surgery

Various robotic platforms have been developed for use in laparoscopic surgery. The first robot approved by the Food and Drug Administration (FDA) for use in

laparoscopic surgery was the Automated Endoscopic System for Optimal Positioning (AESOP). This voice controlled robot provides a stable camera platform for use in laparoscopic surgery [7].

The da Vinci Surgical Robotic System (Intuitive Surgical) is the most prevalent, commercially available surgical robotic device. This system manipulates laparoscopic tools inserted through several laparoscopic ports with the use of arms hovering above the patient. This device is controlled by a surgeon located at a remote workstation. Advantages of this system include dexterous wrist actuation, tremor filtering, and more intuitive controls [8-10]. However, the large size and cost of the da Vinci provide limits from more widespread implementation due to financial restrictions of medical facilities.

Other research is being performed to develop additional robots to control long laparoscopic tools outside the body. Examples include the Raven [11] and COBRASurge [12]. These systems are both more compact and cost effective than the da Vinci. All three of these systems have their tools constrained by incisions in the peritoneum. Because of this restriction, working in multiple quadrants inside the peritoneal cavity can be difficult and time consuming, making these systems impractical for surgeries requiring large, multi-quadrant workspaces, such as colectomies.

Section 2.4: Robots for LESS Surgery

In attempts to solve some of the difficulties associated with single incision procedures, other robotic solutions for LESS are being developed. One instance of this has been accomplished through the use of the da Vinci in single incision procedures instead of the traditional multiport laparoscopic method which it was developed for.

Various surgeries, including a right colectomy, have been performed through using method [13]. These approached have been limited from performing multiple quadrant procedures due to the period of time required to reposition the da Vinci to other quadrants.

A two armed miniature surgical robot, Single-Port lapaRoscopy bImaNual roboT (SPRINT) is also currently being developed for single incision surgery [14]. Both six DOF arms are inserted through a port in the umbilicus. This platform utilizes a combination of both motors and cable-driving for both internal and external actuation. While preliminary benchtop studies have been performed, *in vivo* studies have not yet been performed.

Section 2.5: *In Vivo* Surgical Robots

Previous robotic designs have aimed to overcome the constraints of working through incisions by placing the entire robot inside the peritoneal cavity to assist in laparoscopic surgery. One such example is a pan and tilt camera to provide additional views for placing of trocars during the procedure. Its 15 mm diameter allows the pan and tilt camera to be inserted through a standard laparoscopic port [15-16]. Other *in vivo* robots have utilized two helical drive wheels to provide mobility for the platform. Once inserted through an incision, these mobile robots have demonstrated the ability to traverse the peritoneal cavity. Various versions of this platform have been equipped with tools to allow for cauterizing, clamping, stapling, biopsy, and sensory capabilities [17].

More recent work within this research group have demonstrated the feasibility of a two armed miniature *in vivo* surgical robot capable of performing surgery utilizing a

single incision. Previous platforms of this nature have performed multiple cholecystectomies (gallbladder removals) and a partial colon resection in open procedures. Additionally, the ability to insert a two-armed robot through a single incision and grossly reposition the robot to perform tasks in multiple quadrants has been demonstrated [18].

Chapter 3: Conceptual Design

Section 3.1: Overview

A miniature *in vivo* surgical robotic platform would be very beneficial for use in LESS surgeries. While its application would be useful in a wide variety of surgeries currently performed through standard laparoscopy, the largest impact would be for use in colon resection. Colon cancer is the second most deadly form of cancer, with an estimated 51,000 deaths resulting from colorectal cancer in 2010 [19]. To perform a laparoscopic colon resection, five small incisions are made to allow the introduction of various laparoscopic tools through the peritoneal wall. Once the peritoneal cavity is insufflated with CO_2, these tools are then used by a surgeon to mobilize the colon. Once the cancer-containing section of colon is mobilized, a 50 - 100 mm incision is made to allow the mobilized colon to be pulled through, allowing for the cancerous area to be removed. The remaining unaffected portion of the colon is then reconnected before being returned through the 50 - 100 mm incision.

Due to the shape of the colon and the fact that it spans multiple quadrants in the peritoneal cavity, it is difficult to correctly position the laparoscopic ports to have the proper workspace required to reach multiple quadrants. Currently, out of the 240,000 colon resections performed in the United States annually, less than 20% are performed laparoscopically. Because of this, the vast majority of colon resection surgeries are performed as open procedures, resulting in a 4-6 day hospital stay, coupled with increased discomfort and a larger scar for patients.

To mitigate the issues associated with current colon resection, a robotic platform would be completely inserted into the peritoneal cavity through a single 50 mm incision at the umbilicus, eliminating additional incisions for laparoscopic tools. A central assembly rod would protrude out of the single incision. The robot would then be used to mobilize the colon before the robot is removed through the single incision through which it was inserted. The mobilized colon would then be pulled through this same incision where surgeons could externally remove the cancerous section of colon and reconnect the healthy sections before the colon is reinserted into the cavity through the single incision. This could reduce the hospital stay from 4-6 days required for an open procedure to 1-2 days. This decreased recovery time would also significantly reduce costs associated with extended hospital stays [1]. If the incision is made around the umbilicus, the patient will be left with a "virtually scarless" procedure.

Section 3.2: Design Concepts

Using knowledge gained from previous prototypes created through this research group, this robot will comprise of two arms to perform surgical tasks. Each arm will be individually inserted through a single incision in the peritoneum before being mated together using a central assembly rod. This rod will protrude out the body to allow for external gross repositioning of the robot. This robot will have six degrees of freedom in each arm, as well as the ability for open/close actuation of the end effector. In order to be inserted through the single incision, a maximum of 30 mm equivalent diameter is used as a guideline.

Each arm of the robot will be made up of a torso, upper arm, lower arm, and forearm, as shown in Figure 2. A one degree of freedom shoulder will be located between the torso and the upper arm. The two degree of freedom elbow will be located between the upper arm and lower arm. A two degree of freedom wrist will be located between the lower arm and forearm. The sixth degree of freedom will be a decoupled roll of the end effector and have no effect on the end effector position in Cartesian coordinates. Despite control difficulties associated with controlling a five degree of freedom robot in previous iterations, control will be simplified through the implementation of an approximate three degree of freedom "wrist" between the lower arm and forearm. With this addition, the inverse kinematic solution for the Cartesian coordinates of this approximated wrist will be calculated geometrically using the first three degrees of freedom of the robot. The final three degrees of freedom will be completely decoupled from each other, where each degree of freedom provides the rotational orientation of the end effector positioned at the end of the forearm.

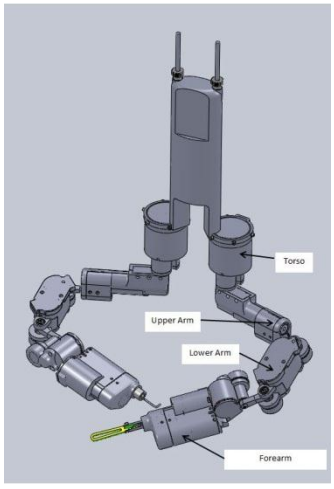

Figure 2: Assembled miniature surgical robot platform

Section 3.3: Design Requirements

Various design constraints that are necessary for an *in vivo* platform designed for LESS often are conflicting and require trade-offs. The largest constraint is the size of each arm of the robot, as it needs to be able to be inserted through a single incision into the peritoneal cavity without insufflation. Because of this, the target equivalent diameter of the robot is constrained to approximately 30 mm. This size is the main driving force for most of the design decisions. Once the robot is inserted inside the cavity, it needs to have the necessary workspace, dexterity, force, and speed required to perform surgical tasks.

Researchers at the BioRobotics Lab at the University of Washington have performed studies to quantify the forces and speeds used by surgeons performing common surgical tasks such as running the bowel, dissecting mesenteric arteries, passing a suture, tying a knot, and suturing colon. To measure the forces, the BlueDRAGON system was constructed to measure the forces applied by the surgeon while performing these laparoscopic tasks in the peritoneal cavity. It must be noted that the forces applied by a surgeon are not the same forces applied to the tissue *in vivo* as a result of the friction in the trocar and the reaction with the abdominal wall. Due to the limited information currently available on force requirements for *in vivo* tissue manipulation, these force values can be used as a preliminary design goal [20-21].

A decision must be made to maximize the usable workspace for the robot while limiting the overall length to allow for insertion into the noninsufflated peritoneal cavity. Additionally, intersecting workspace must be considered, as it is often required that both

end effectors be capable of reaching tissue simultaneously to perform cooperative surgical tasks. Through the kinematic orientation of the joints chosen for the design, the robot is capable of maintaining dexterity in a large workspace spanning from areas close to the torso of the robot out to the extents of the cavity. Based on initial work, this robot is capable of performing surgical tasks in all four quadrants without external gross repositioning of the robot.

Chapter 4: Kinematic Design

Section 4.1: Denavit-Hartenberg Parameters

A kinematic model of the six DOF right arm of the LESS surgical robot is shown in Fig. 3. The base frame {0} and frame {1} are located at the one DOF shoulder joint with the z-axis of both frames along the shoulder actuation shaft. Frames {2} and {3} intersect at the two DOF elbow joint located between the upper arm and lower arm. Frames {4} and {5} are located between the lower arm and forearm. The end effector rotational frame, {6}, is located along the end of the forearm.

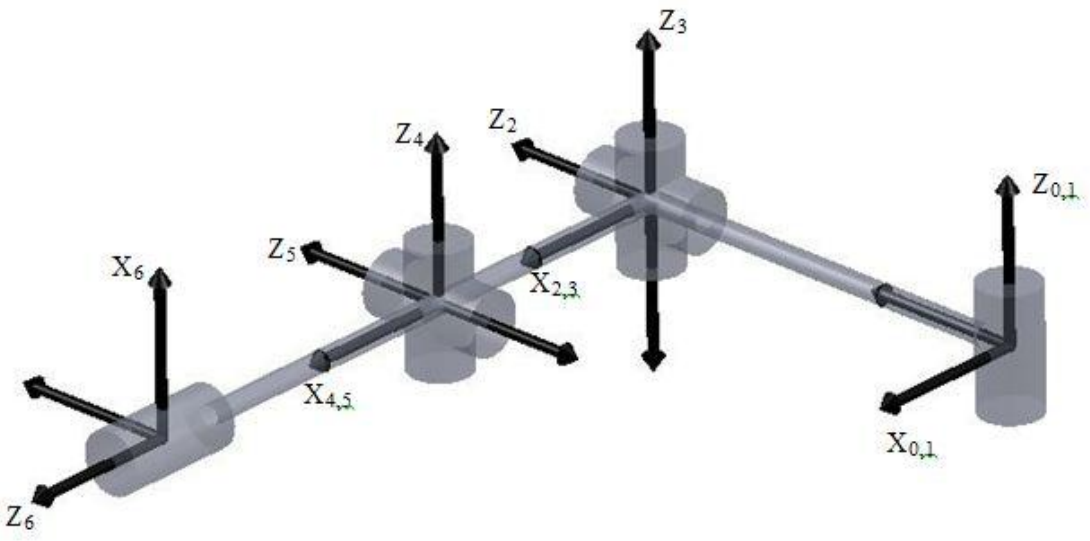

Figure 3: Six DOF kinematic model

The Denavit-Hartenberg parameters for the right robotic arm are displayed in Table 1. The sixth reference frame is completely decoupled and not included in Table 1. In this table, L_{UA} is the length of the upper arm, L_{LA} is the length of the lower arm, and L_{FA} the length of the forearm.

Table 1: Denavit-Hartenberg parameters

i	α_{i-1}	a_{i-1}	d_i	θ_i	θ Limits
1	0	0	0	Θ_1	-135 – 135
2	-90	0	$-L_{UA}$	Θ_2	-180 – 180
3	90	0	0	Θ_3	-90 – 90
4	0	L_{LA}	0	Θ_4	-90 – 90
5	90	0	0	Θ_5	-90 – 90

Section 4.2: Forward Kinematics

Once the Denavit-Hartenberg parameters are determined, they can be used to determine the frame transformations of each frame from the relationship

$$ {}^{i-1}_{i}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

resulting in

$$ {}^{0}_{1}T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$^{1}_{2}\mathrm{T} = \begin{bmatrix} c_2 & -s_2 & 0 & 0 \\ 0 & 0 & 1 & -L_{UA} \\ -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{2}_{3}\mathrm{T} = \begin{bmatrix} c_3 & -s_3 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{3}_{4}\mathrm{T} = \begin{bmatrix} c_4 & -s_4 & 0 & L_{LA} \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{4}_{5}\mathrm{T} = \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{5}_{6}\mathrm{T} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -s_6 & -c_6 & 0 & 0 \\ c_6 & -s_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

For simplification, the link length between the lower arm and forearm is approximated as zero due to its relative short length. Since the sixth degree of freedom is decoupled from the end effector position, its transformation matrix was found without using the Denavit-Hartenberg parameter. The Cartesian coordinates of the end effector can be found using the forward kinematic transform

$$P = \begin{bmatrix} \Big(\big((\mathrm{c_1c_2c_3} - \mathrm{s_1s_3})\mathrm{c_4} + (-\mathrm{c_1c_2s_3} - \mathrm{s_1c_3})\mathrm{s_4}\big)\mathrm{c_5} + \mathrm{c_1s_2s_5}\Big)\mathrm{L_{FA}} + (\mathrm{c_1c_2c_3} - \mathrm{s_1s_3})\mathrm{L_{LA}} + \mathrm{s_1L_{UA}} \\ \Big(\big((s_1c_2c_3 + c_1s_3)c_4 + (-s_1c_2s_3 + c_1c_3)s_4\big)c_5 + s_1s_2s_5\Big)L_{FA} + (s_1c_2c_3 + c_1s_3)L_{LA} - c_1L_{UA} \\ \big((-s_2c_3c_4 + s_2s_3s_4)c_5 + c_2s_5\big)L_{FA} - s_2c_3L_{LA} \end{bmatrix}$$

Section 4.3: Workspace

Due to the necessity for the robot to have the capabilities to perform surgical tasks in multiple quadrants, as required for colectomies, the kinematic orientation and joint limits of the robot allow for a dexterous workspace spanning four quadrants of the peritoneal cavity. This orientation provides for a wide range of motion both close to the torsos of the robot as well extending to the outer edges of the cavity. The majority of the workspace consists of an inner spherical boundary with a radius of 41 mm and an outer boundary of an approximate torus with a tube diameter of 105 mm located 65 mm from the torus center. Due to the placement of the second arm's torso, this workspace shape can be achieved in three of the four quadrants. The workspace for the remaining quadrant is reduced in volume, yet can likely provide sufficient reach and dexterity to perform surgical tasks in that quadrant.

To perform many surgical tasks, the intersecting workspace is a better measure of the capabilities of the robot. In Figures 4 and 5, the left and right arms are represented by the blue and red tori, respectively, while the purple volume represents the intersecting workspace. This demonstrates the ability of the robot to perform procedures in all four quadrants of the peritoneal cavity. A SolidWorks rendering of this capability is shown in Figure 6. When the robot is working in the quadrant opposite of the initial orientation, the end effector's orientation will switch, resulting in the left end effector being on the right side and vice versa. Additionally, the ability of multiple quadrant manipulation was demonstrated on a benchtop experiment, as shown in Figure 7.

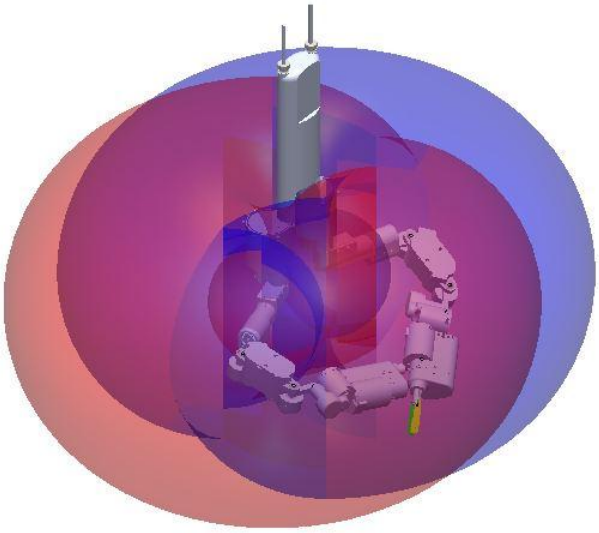

Figure 4: Model of intersecting workspace with surgical robot

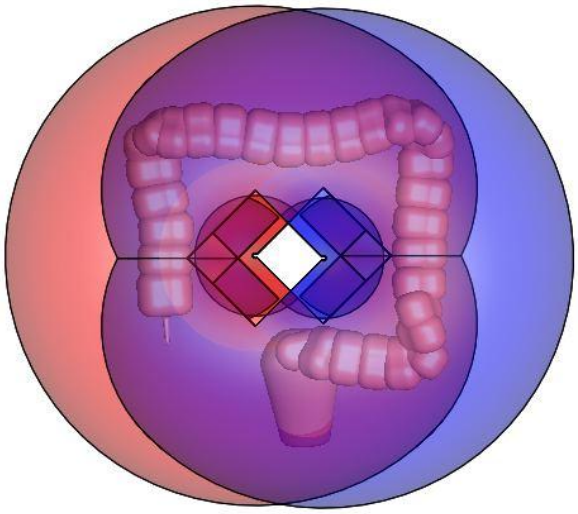

Figure 5: Model of intersecting workspace with respect to the colon

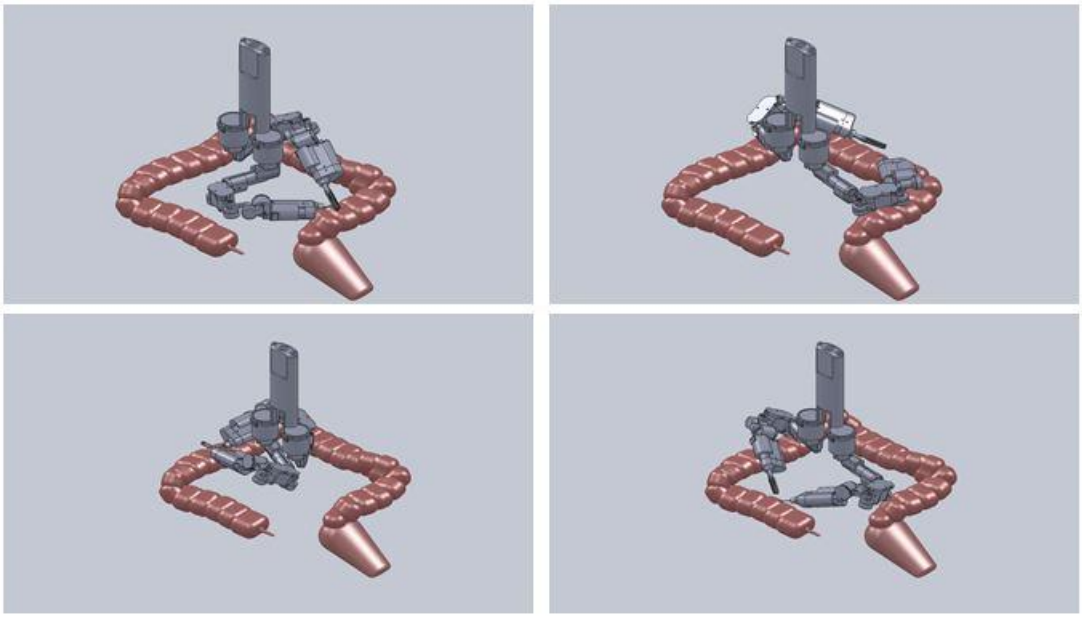

Figure 6: Model of surgical robot's four quadrant capabilities

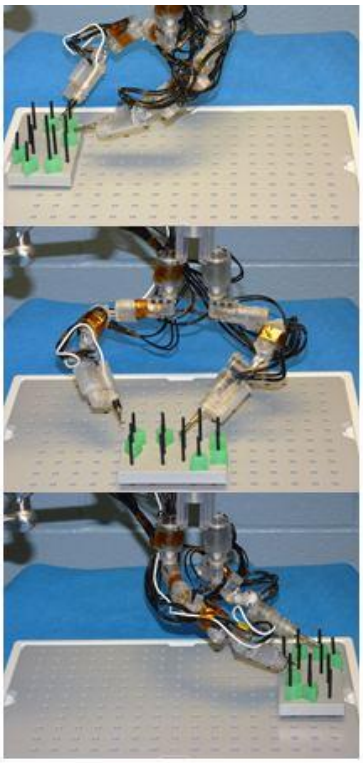

Figure 7: Benchtop experiment of multiple quadrant capabilities

Section 4.4: Jacobian

The Jacobian matrix can be utilized to determine information such as manipulability, force, and speed capabilities of the robot based on its kinematic and joint properties. The infinitesimal translation and rotation of the end-effector corresponds with the manipulator Jacobian. The Jacobian matrix for this six DOF robot would take the form of a 6 x 6 matrix made up of column vectors representing the velocity and linear velocity generated by each corresponding joint. This method of computing the Jacobian follows the formula of

$$J = \begin{bmatrix} J_L \\ J_A \end{bmatrix}$$

where each component is the sum of

$$J_{L,i} = b_{i-1} \, x \, r_{i-1,e}$$

and

$$J_{A,i} = b_{i-1}.$$

In this equation, b_i is the unit vector for the joint and $r_{i\text{-}1,e}$ is the position vector [22]. Calculation of components of the Jacobian matrix was performed utilizing MATLAB. The code used is further detailed in Appendix A, along with the output of the 6 x 6 Jacobian matrix.

The 6 x 6 Jacobian matrix can be used to determine the end effector force capabilities of at various positions across the workspace. Joint torques are related to end effector forces by the equation

$$\tau = J^T F$$

where τ is the joint torque vector, J^T is the transpose of the Jacobian matrix, and F is the force vector.

Using the guidelines based on the Blue Dragon data of maximum required forces in the X, Y, and Z direction of 5 N, 5N, and 20 N, respectively, the force capabilities were tested. Although of the values of the Blue Dragon were force requirements required by a user controlling laparoscopic tools from outside the body where trocar friction and lever arms effect the end effector forces, they can provide initial target values. Due to the fact the workspace of the robot is revolved, a cross section of the workspace represents the force capabilities of the robot across the majority of the workspace. MATLAB was utilized to plot the data points in this cross sectional plane where the respective values of F_x, F_y, and F_z met the requirements the force targets of 5 N, 5 N, and 20 N, respectively, as shown in Figure 8.

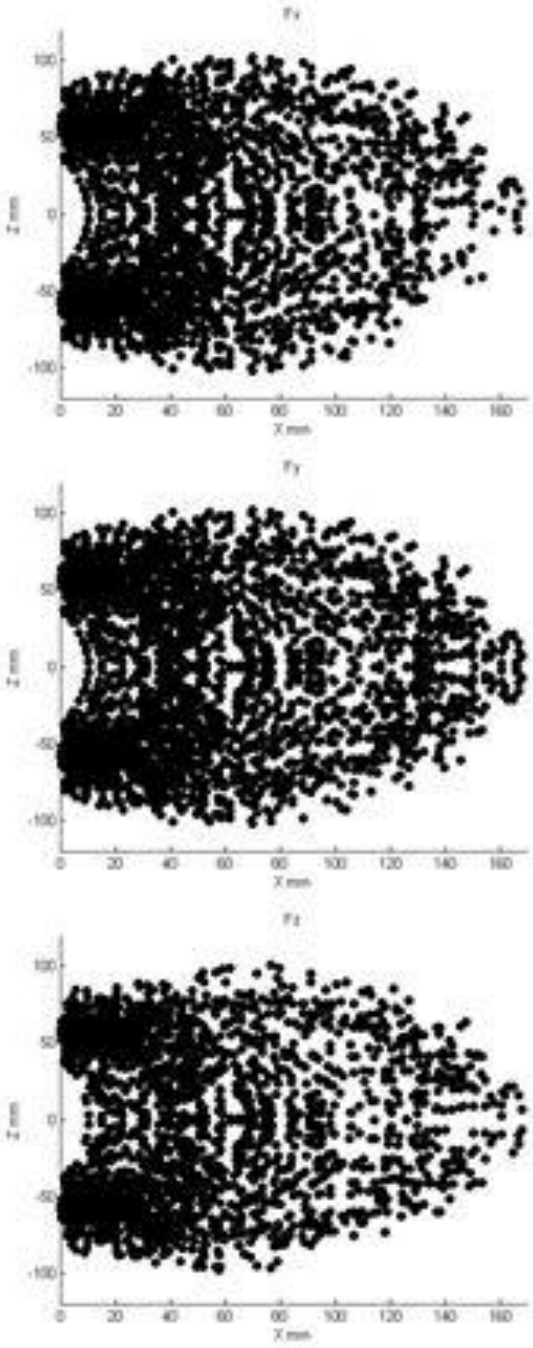

Figure 8: Cross sectional force data of the surgical robot's ability to meet or exceed force requirements determined by the BlueDRAGON

The first three joints were stepped through their joint limits with a step size of 10˚ while the last three joints were stepped through their joint limits with a step size of 5˚. The cross section used for plotting was .5 mm thick centered at the y = 0 plane. These

plots demonstrate the theoretical force capabilities meet or exceed the requirements as determined by the Blue Dragon over a wide range of the workspace. With a smaller step size, the addition data points would likely fill in more of the workspace area.

The velocity capabilities of the robot at various points across the workspace can also be found by utilizing the Jacobian matrix. The equation relating end effector angular velocity is

$$V = J\dot{\theta}$$

where V is the linear velocity vector and $\dot{\theta}$ is the joint angular velocity vector. In a similar method as for plotting force, MATLAB was utilized to plot data points where the robot was capable of obtaining a maximum linear velocity of the end effector benchmark of 0.072 m/s, as found through the BlueDRAGON data. A step size of 10˚ was used for all six joints. Each data point represents a point in the robot workspace where a linear velocity of 0.072 m/s is obtainable in the 1 mm cross section centered at the y = 0 mm plane. These plots are displayed in Figure 9.

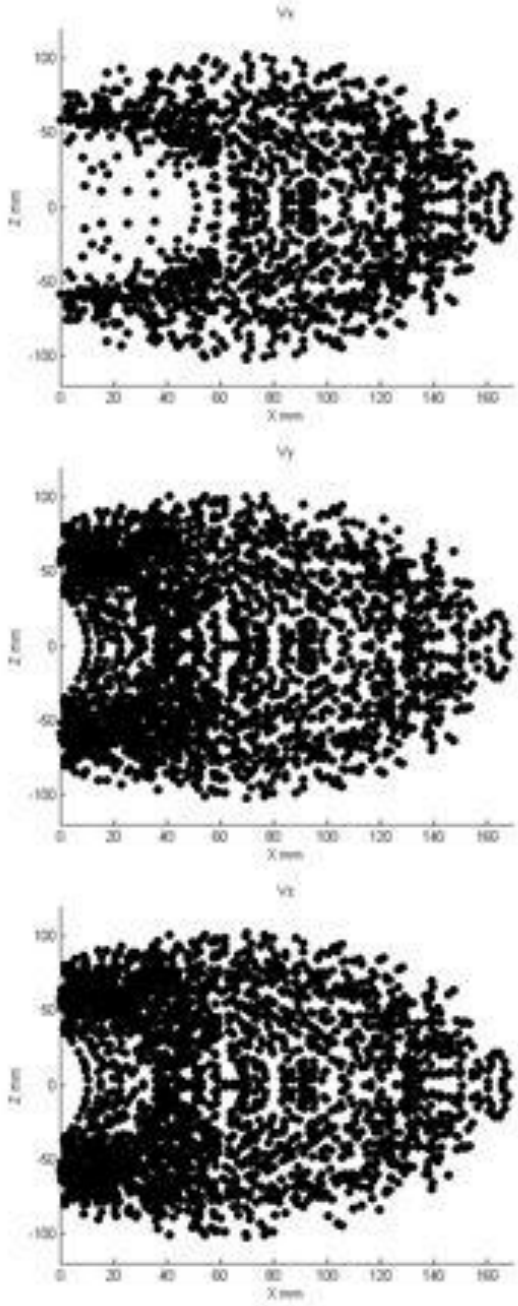

Figure 9:Cross sectional velocity data of the surgical robot's ability to meet or exceed linear velocity requirements determined by the BlueDRAGON

Chapter 5: Prototype LESS Robot

Section 5.1: Design

A miniature *in vivo* surgical robot was developed for insertion through a single incision into the peritoneal cavity to perform surgical tasks. The robot, as shown in Figure 10, consists of two arms, both containing a torso, upper arm, lower arm, and forearm. Once inserted inside the peritoneal cavity, both arms are mated together with the use of a central assembly rod. The left and right end effectors are a grasper and a monopolar hook cautery, respectively. The end effectors are interchangeable depending on the procedure requirements.

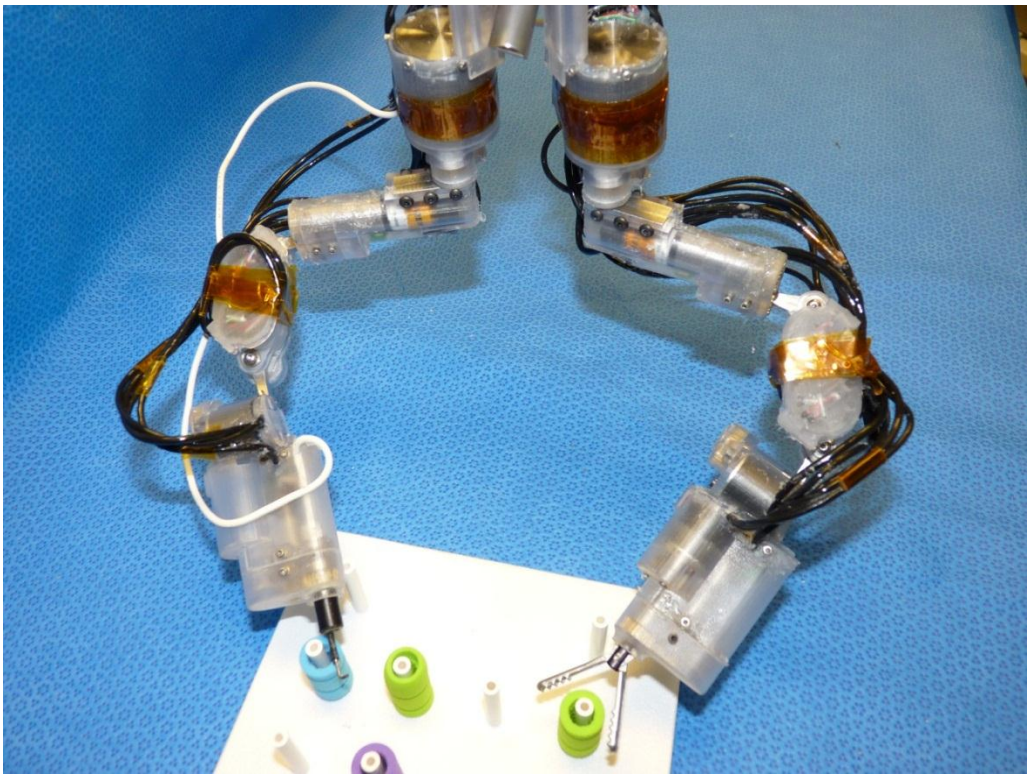

Figure 10: Prototype surgical robot

Section 5.2: Insertion and Attachment

Each arm of the robot is independently inserted through a single incision in the peritoneum from its home position. Once each arm of the robot is individually inserted into the peritoneal cavity through the single incision in the peritoneum, one control rod rigidly attached to each torso will be left protruding through the incision. Each of these control rods are aligned with the corresponding hole before the assembly rod is lowered until coming in contact with the torso of each arm.

Each torso then mates with the assembly rod, being constrained rotationally due to the geometry of each piece. A thumb screw is then threaded onto the free end of each control rod until the translation of each torso is constrained to the assembly rod. A rubber gasket between the thumb screw and the assembly rod prevents the escape of CO_2 used for insufflation. The free end of the assembly rod is fed through a Gel Port and extra wire is pulled through the wire races. The Gel Port is then attached to the lower ring and the peritoneal cavity is insufflated.

Section 5.2: Electronics and Communication

The communication and power for the actuation of the robot is provided through tethered wires connected to each motor. Each motor is a coreless permanent magnet direct current motor with a magnetic encoder. These wires are connected to CompactRIOs (National Instruments) NI 9505 motor modules, which provide motor driving using a proportional-integral-derivative (PID) control method. Passive filter printed circuit boards (PCB) have been implemented between the motors and the CompactRIO to filter out signal interference that may be introduced due to the electrical

signal from the cautery. Power for both the motor and encoders are provided using external power supplies. An Ethernet cable facilitates communication between the CompactRIOs and a laptop computer. Mcobotor position information is sent to the CompactRIOs from LabVIEW graphical programming environment (National Instruments).

Section 5.3: Remote Surgical Interface

The remote surgical user interface system consists of a monitor, triple-action footpedal, and controller. Two controllers were developed in parallel. The first system utilizes Phantom Omni (Sensable) haptic devices while the second system utilizes a scaled, kinematically-matched Master.

5.3.1: Master Controller

One of the controllers for the remote surgical user interface system developed used a kinematically matched master that was scaled 1.8:1 as compared to the surgical robot. The master system consists of two arms mounted onto an adjustable base. Each arm has a one degree of freedom shoulder, two degree of freedom elbow, two degree of freedom wrist, and handle that can be rolled. Additionally, a trigger is used to control the actuation of the end effector. As the handles, which represent the end effector of each robotic arm, are moved around the workspace of the master, potentiometers at each joint record the joint angles for each arm. This information is then sent to LabVIEW via a USB DAQ (National Instruments) to provide motor position set points for each joint of the robot. Since each joint is directly mapped, there is no need to calculate the inverse

kinematic solution for each end effector position. An image of the Master is shown in Figure 11.

The monitor provides visual feedback for the surgeon. The triple action foot pedal provides for locking of both the right and left arm of the surgical robot. Before each arm of the robot is unlocked, the respective arm of the master must be positioned in the same orientation as the robot to avoid undesired movement of the robot.

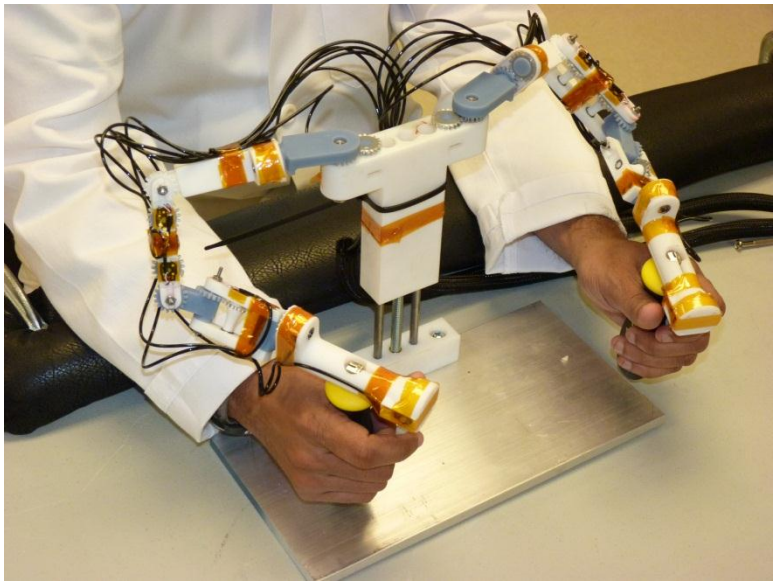

Figure 11: Master controller for remote surgical interface

5.3.2: Phantom Omni Controller

A second control system was implemented utilizing two Phantom Omni (Senasable) haptic devices with specialized handles, as shown in Figure 12. The

Phantom Omni devices have the capability of measuring six DOF. The X,Y,and Z Cartesian coordinates and three rotational degrees of freedom of the devices gimbal are measured. The device also provides force feedback in the three translational degrees of freedom. To use this device for the six DOF robot, the robot was broken into two sets of three degrees of freedom. The first system is determined by the Cartesian coordinates of the wrist joint. The rotation of the gimbal is then used to correspond to the three rotational degrees of freedom (Rx, Ry, and Rz) of the forearm of the robot. By doing this, the problems associated with multiple solutions to larger order degree of freedom robotic systems are mitigated. It is approximated that the three axese intersect at the wrist despite the length of the link between the lower arm and forearm. This is justified due to the relatively short length of this link and the fact that the robot is controlled based on the vision of a surgeon, allowing for intuitive compensation by the surgeon to position the end effector. This visual compensation is also used to mitigate the effects of motor backlash propagation through the robot, although solutions to reduce backlash are being attempted.

Figure 12: Phantom Omni controller for remote surgical interface

The position of the wrist can be determined based on the first three rotational frames of reference corresponding to the first three joints of the robot as

$$P_{wrist} = T_1^0 T_2^1 T_3^2 P$$

where

$$P = \begin{bmatrix} L_{LA} \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

This resulting forward kinematic location of the wrist joint gives the Cartesian coordinates based on the Phantom Omni coordinate system of

$$\mathrm{X} = (c_1 c_2 c_3 - s_1 s_3) L_{LA} + s_1 L_{UA}$$

$$\mathrm{Y} = (s_1 c_2 c_3 + c_1 s_3) L_{LA} - c_1 L_{UA}$$

$$\mathrm{Z} = -s_2 c_3 L_{LA}$$

In order to map the joint angles for a desired wrist position, the inverse kinematic solutions for each of the joints can be solved using the geometric solution. The inverse kinematic solutions for the first three angles follow the equations

$$\theta_1 = abs(\arctan\left(\frac{x}{z}\right)) - \arccos(\frac{L_{LA}^2 - z^2 - (x^2 + y^2) - L_{UA}^2}{-2 * L_{UA} * \sqrt{x^2 + y^2}})$$

$$\theta_2 = 1.570796 - \arccos(\frac{L_{UA}^2 + L_{LA}^2 - |P_{wrist}|^2}{2 * L_{UA} * L_{LA}})$$

$$\theta_3 = \arcsin(-\frac{z}{L_{LA}\cos(\theta_2)})$$

The position of the wrist is prevented from exiting the usable workspace through the use of haptic feedback in the Cartesian coordinates of the Phantom Omni controllers. A haptic workspace consisting of a torus and inner cylinder was programmed into a C++ program used to communicate between the Phantom Omni controllers and the LabVIEW program. The C++ code is attached in the Appendix A.

A custom handle was designed to be implemented with the Phantom Omnis to control the end effector position and orientation on the robot. The handle is scaled such that the length from the gimbal axis to the end of the handle corresponds to the length between the wrist of the robot and the end effector. Once the handle is in the surgeon's hand and the robot is unlocked, the Rx, Ry, and Rz orientations of the handle will be translated to the corresponding orientations for the robot's end effector. While Rz is completely decoupled from the other joint angles of the robot, Rx and Ry are partially

decoupled from the other joint angles. Due to the rotational orientation of the lower arm, each of these angles must be compensated for, resulting in the last three joint angles to be calculated from

$$\theta_4 = Rx - (90 + \theta_1 - \arccos(\frac{-(L_{LA}^2 - z^2) + x^2 + y^2 - L_{UA}^2}{-2 * L_{UA} * \sqrt{L_{LA}^2 - z^2}})$$

$$\theta_5 = Ry - \theta_3$$

$$\theta_6 = Rz$$

This solution mitigates previous issues resulted from the lack of haptic feedback in the gimbal which could be used to prevent the robot from reaching orientations outside of its joint limits which were possible to reach with the Phantom Omnis. It should be noted that in the equation for θ_4 the area of the workspace where $L_{LA}^2 - z^2 = 0$ corresponds with the upper boundary of the workspace. Due to the implemented haptic boundary limits, this situation is prevented from occurring. With previous generations of higher degree of freedom robots, when these orientations outside the robot's workspace were reached, an inverse kinematic solution could not be solved, which often resulted in the robot "jumping" into undesired orientations. Through the method of partially decoupling the gimbal orientation, only one joint is affected if the orientation is not within the robot's workspace. If the determined joint angle is out of the range of joint limits for that particular joint, the joint is held at that joint limit until the gimbal is brought back into the usable workspace volume. No other joints are affected by this exiting of the workspace. Additionally, the internal mechanical joint limits of the Phantom Omni controller and the ergonomic arrangement of the user interface greatly

reduce the occurrence of a joint set point exceeding its joint limit. A spring-loaded trigger allows for open/close actuation control of the respective end effector. The angle of the trigger is read through the use of a potentiometer and is mapped to a corresponding angle of the robotic end effector.

This system provides possible advantages over the direct master-slave system, including tremor filtering, clutching, and motion scaling. In order to implement motion scaling, an adjustable handle would need to be implemented to compensate for the change in scale to the forearm. Due to the approximate intersection of the final three axes, Pieper's solution [23] was originally considered as an inverse kinematic solution to use for the controller. Because of complications associated with the lack of rotational orientation haptic feedback, Pieper's solution was not implemented for this controller.

Section 5.4: Segment Design

The design of each arm segment will be presented in further detail. Each of the components for the torso, upper arm, lower arm, and forearm are identical, and therefore interchangeable between the left and the right arm.

5.4.1: Torso

Shown in Figure 13, the torso motor housing holds the motor and actuation mechanism for the first joint. A spur gear is rigidly attached to the output shaft of the torso motor. As the motor output shaft turns, the motor spur gear rotates the output shaft spur gear, which is radially constrained with the torso rotational shaft through a flat placed on the output shaft and the bore of the spur gear. The output shaft is supported with two flanged ball bearings. The lower flanged ball bearing is seated in the lower

torso cap. The output shaft is constrained to the torso to upper arm link with a bolt. A hex nut provides an axial constraint to the output shaft.

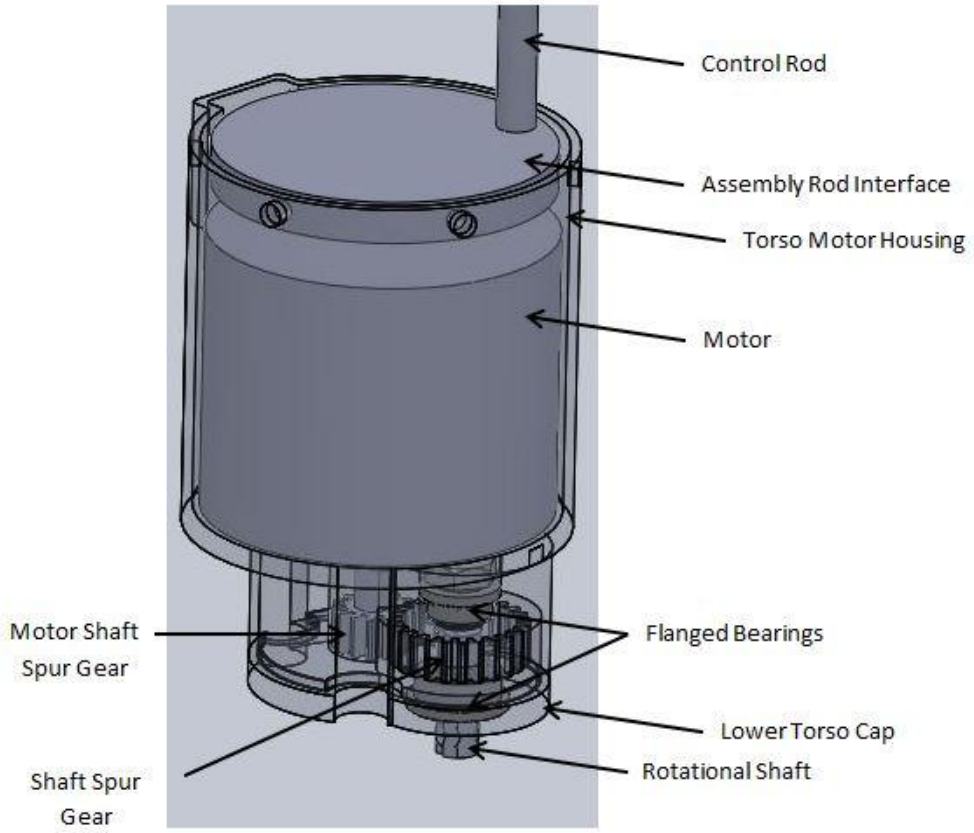

Figure 13: Internal view of the torso

5.4.2: Upper Arm

The torso to upper arm link is constrained to both of the upper arm motor housing halves with bolts threaded into the torso to upper arm link, as shown in Figure 14. The second joint is actuated from a motor located inside the upper arm motor housing. An encoder provides position information for the motor. A planetary gearhead is attached to the motor by way of mating threads standard on these motors and planetary gearheads.

The planetary gearhead is rigidly attached to the upper arm gear housing by use of epoxy to prevent rotation and translation of the motor assembly. A spur gear is rigidly attached to the output shaft of the gearhead. As the spur gear is rotated by the motor, torque is transmitted to the output spur gear, which is rigidly attached to the upper arm to lower arm link. This link is supported by a pair of ball bearings housed in the upper arm gear housing. A button socket cap bolt is threaded into this the link, preventing translation. This joint provides roll for the lower arm.

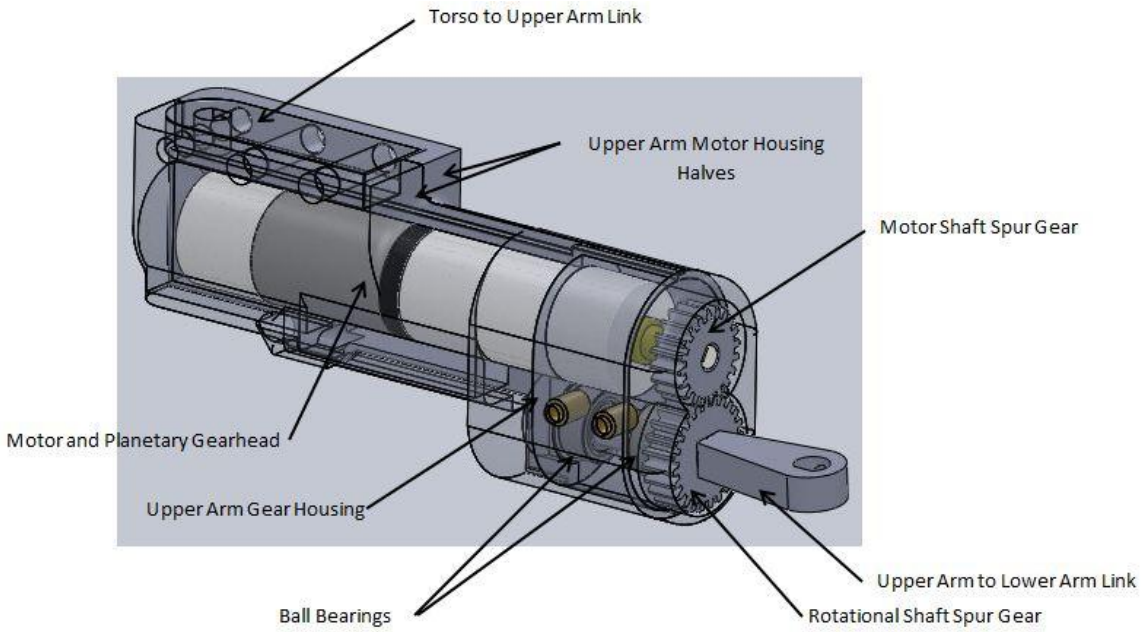

Figure 14: Internal view of the upper arm

5.4.3: Lower Arm

Shown in Figure 15, the upper arm to lower arm link interfaces with the lower arm by way of the lower arm rotational shaft. A flat has been placed on both the lower arm rotational shaft and the bore of the upper arm to lower arm link to constrain the

rotation of these parts. A set screw also threads into the link to constrain the parts axially. Ball bearings, which are housed in the lower arm motor housing, support the lower arm rotational shaft. A spur gear is rigidly attached to the rotational shaft. A bolt constrains the rotation shaft axially. The rotation shaft is rotated as the 15 mm motor is actuated, rotating the motor output spur gear, which is rigidly attached to the output shaft of the motor. The motor is constrained utilizing two bolts which go through the motor housing and are threaded into the mounting holes of the motor. The lower arm gear cover covers the moving gears to prevent outside objects from contacting the moving gears. It is held in place by the mounting bolts for the motor. This joint provides yaw for the lower arm. This joint actuation mechanism is identical to the actuation of the fourth joint with the lower arm to forearm link being actuated with respect to the lower arm, providing yaw for the forearm.

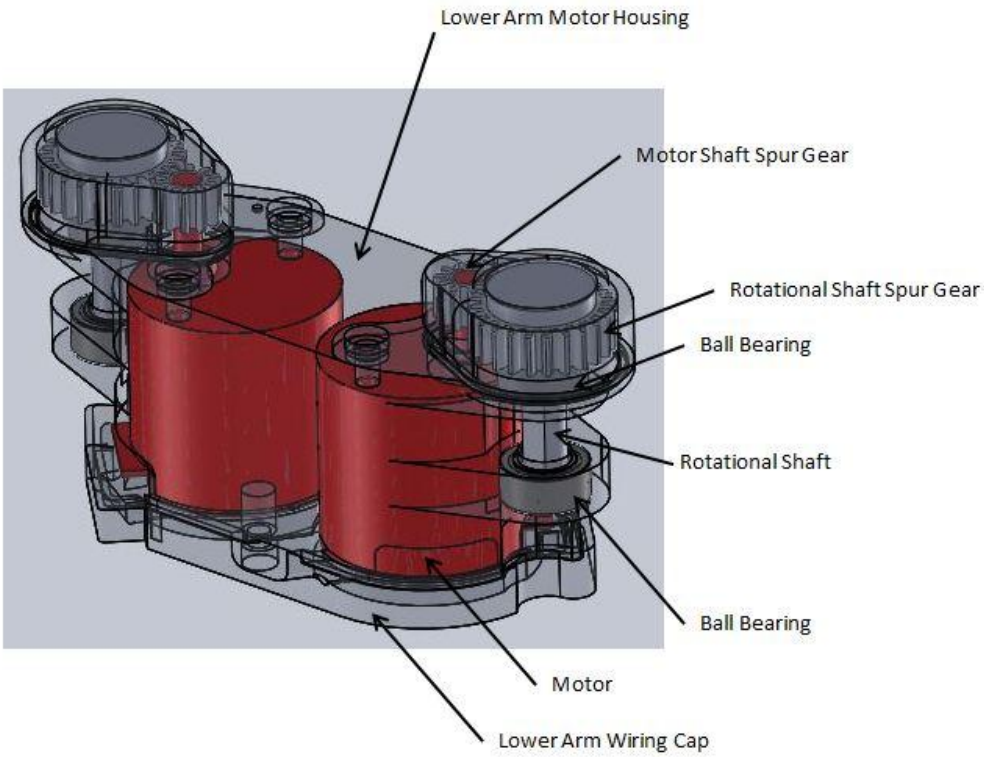

Figure 15: Internal view of the lower arm

5.4.4: Forearm

The internal workings of the forearm pitch joint are shown in Figure 16. The lower arm to forearm link is connected to the forearm by interfacing with the forearm rotational shaft. A flat has been placed on both mating surfaces to constrain these parts radially. The forearm rotational shaft is supported by ball bearings which are seated in the forearm pitch motor housing. A spur gear is rigidly attached to the rotational shaft. The rotational shaft is constrained axially through the use of a button socket cap bolt. When the 15 mm motor is actuated, the motor shaft spur gear, which is rigidly attached to the output shaft of the motor, also rotates. The motor is constrained by two bolts through

the motor housing which are threaded into the mounting holes of the motor. These bolts also hold the forearm gear cover in place to prevent outside objects from contacting the moving gears. Torque is transmitted from the motor spur gear to the rotational spur gear, providing pitch for the forearm. The forearm pitch motor wiring cover provides a back support for the motor as well as covers the wiring. This cover is held in place using two bolts through the motor housing. Four bolts attach this motor housing to the rest of the forearm.

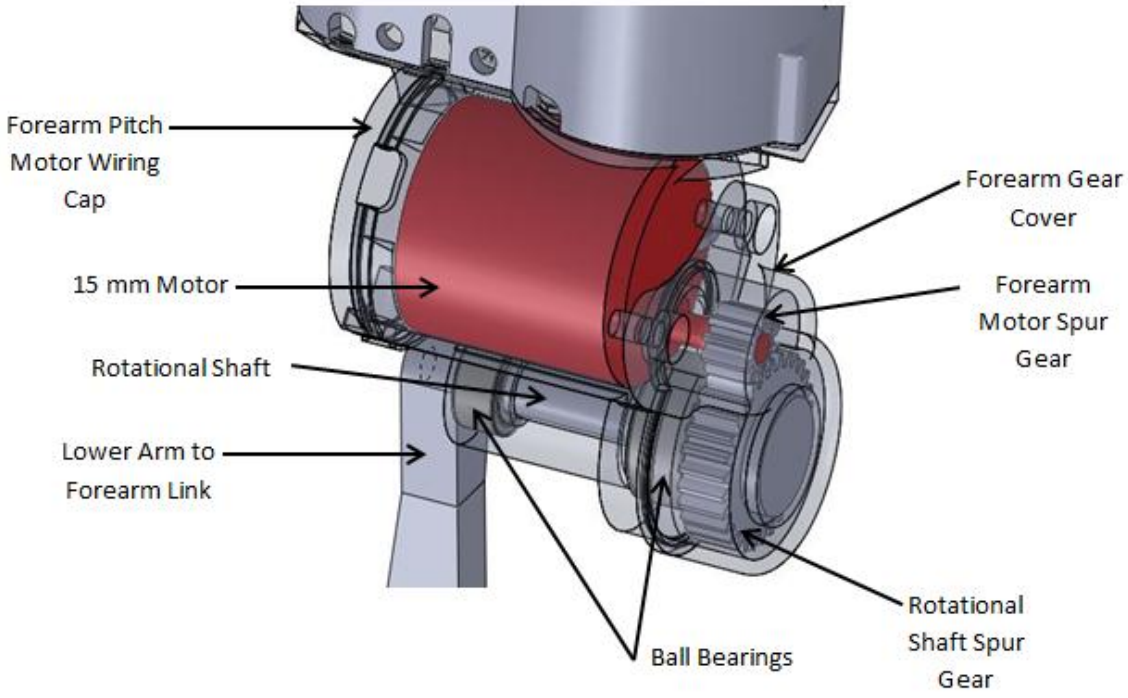

Figure 16: Internal view of the forearm pitch joint

Shown in Figures 17 and 18, the grasper housing is mated with the rotational motor spur gear. Actuation of the rotational motor and rotational motor gearhead causes

rotation of the rotational motor spur gear, also resulting in the rotation of the grasper housing. The grasper housing is supported by two bearings to reduce rotational friction of the grasper housing. A distal hex preload nut limits translation of the grasper housing and provides a preload for the bearings to further help reduce friction during rotation of the grasper housing. A beveled washer is located between the ball bearing and hex preload nut to provide compliance.

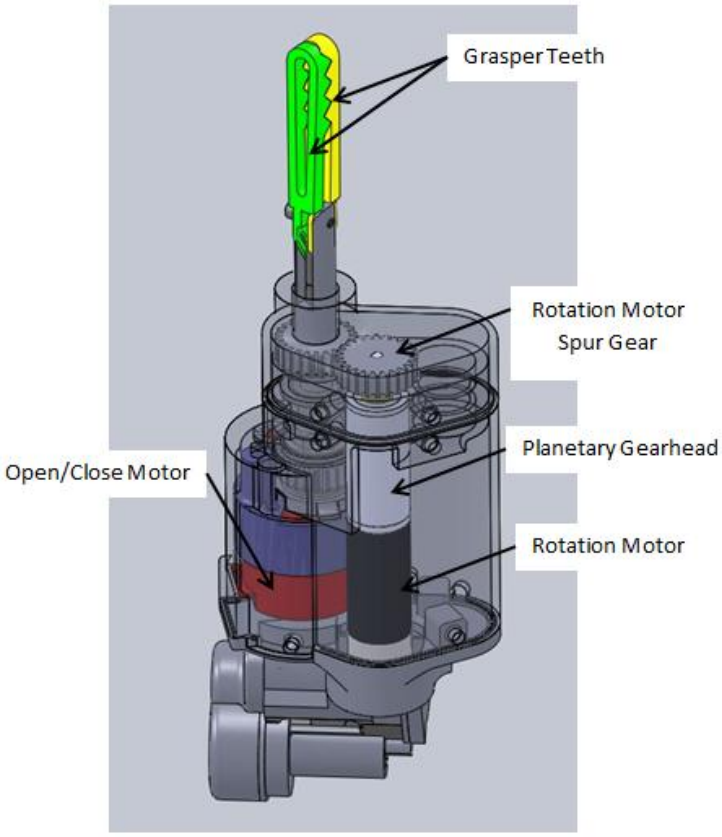

Figure 17: Internal view of grasper forearm

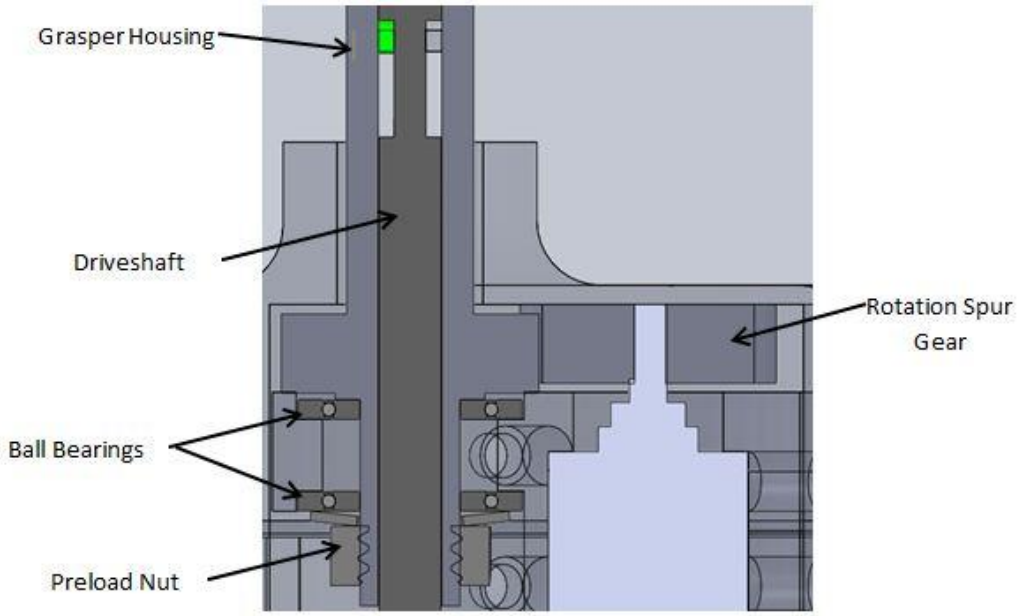

Figure 18: Cross-sectional view of grasper rotation mechanism

The grasper actuation motor is rigidly coupled to the motor housing by two actuation motor mounting bolt, as shown in Figure 19. These two mounting bolts constrain the translation and rotation motion of the actuation motor to motor housing. The motor is rigidly attached to a spur gear. Actuation of this motor causes rotation of the spur gear, which leads to the rotation to the driveshaft housing spur gear. The driveshaft housing spur gear is rigidly coupled to the driveshaft housing coupled to the grasper driveshaft. Rotation of the driveshaft housing spur gear is provided through the actuation motor, resulting in rotation of the driveshaft housing and the translation of the grasper driveshaft due to it being constrained radially by grasper teeth.

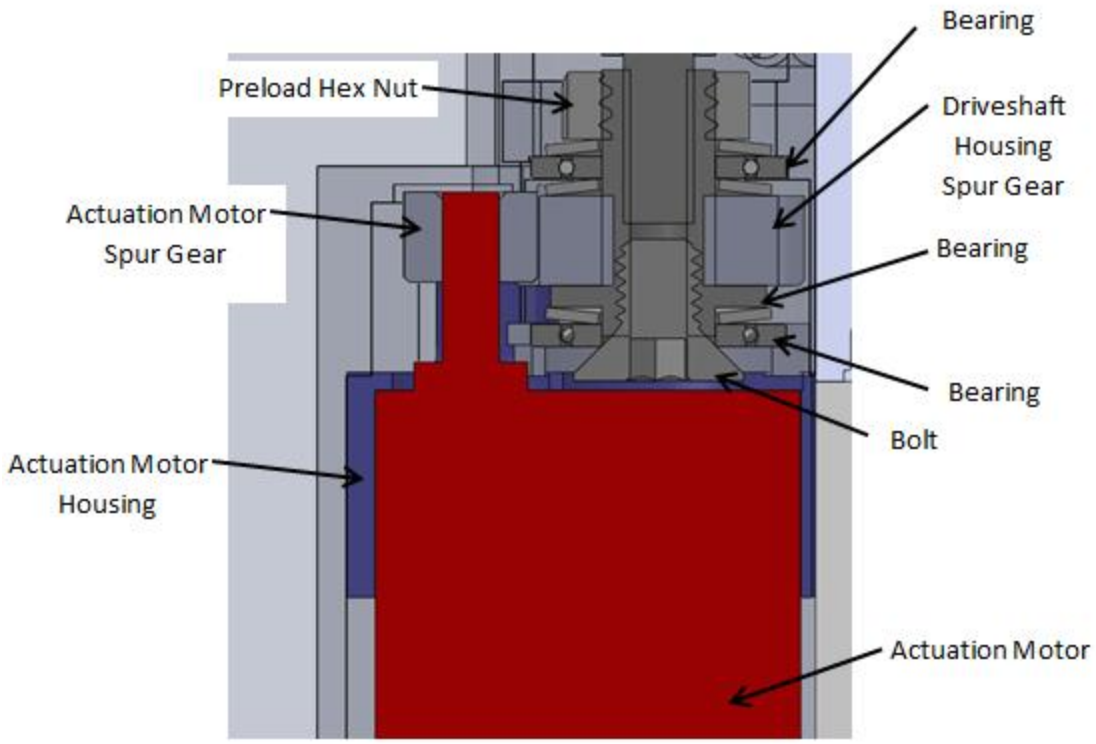

Figure 19: Cross sectional view of grasper driveshaft actuation mechanism

As shown in Figure 20, a grasper rotation bolt threads through one side of the grasper housing and extends through a hole in both grasper teeth. A pin machined into the grasper driveshaft rides in the grooves of the grasper teeth. As the grasper driveshaft is translated forward and back, the pin moves along the grooves, resulting in the opening and closing of the graspers.

The rotation mechanism of the grasper driveshaft is assisted by a proximal hex preload nut, beveled washers and bearing elements. The driveshaft housing is rigidly coupled to a driveshaft housing screw. This constrains translation of the driveshaft housing to the proximal bearing.

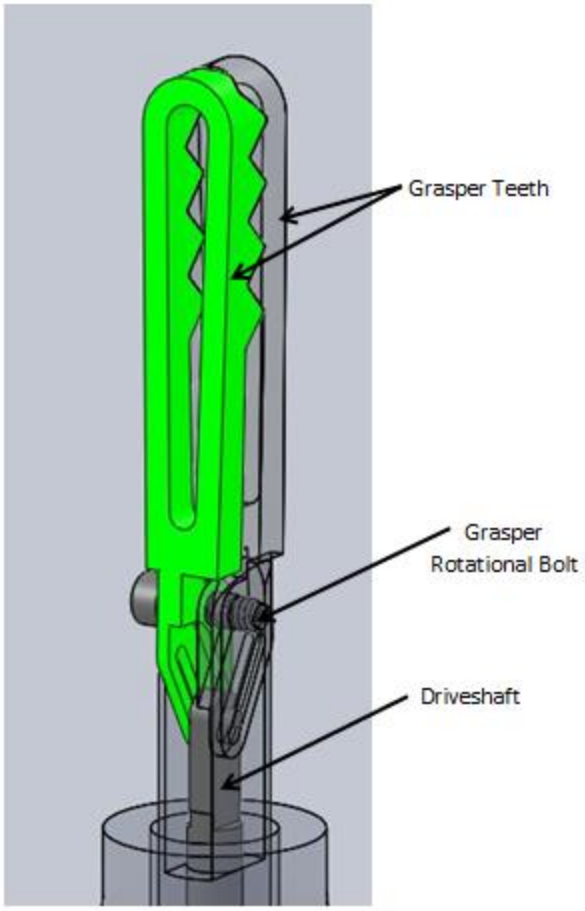

Figure 20: View of grasper teeth actuation mechanism

The cautery rotational gear is mated with a rotational motor spur gear, as shown in Figure 21. The motor spur gear is actuated by a rotational motor and motor gearhead coupled to the motor. Actuation of this motor and gearhead causes rotation of the motor spur gear, resulting also in the rotation of the cautery rotational gear and the cautery housing. The cautery housing is supported by two bearing elements proximal to the cautery rotational gear: The cautery housing and proximal bearing are further coupled to a cautery shaft nut that limits translation of the cautery housing and provides a preload

for the two bearing elements to aid in reducing friction during rotation of the cautery shaft.

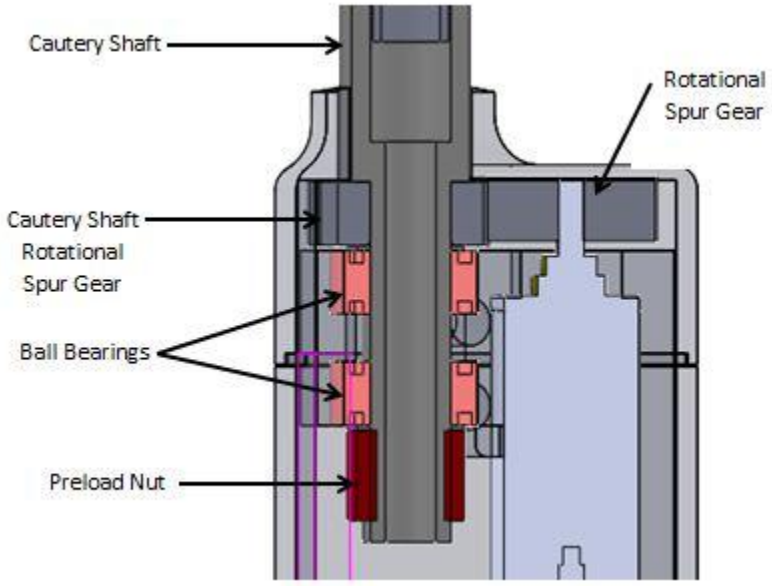

Figure 21: Cross-sectional view of cautery rotation mechanism

Section 5.5: Dual End Effector Forearm

This robot is also capable of utilizing a dual end effector forearms for use with bi-directional kinematics on a miniature *in* vivo surgical robot. In this design, two end effectors can be utilized by a surgeon controlling the miniature *in vivo* surgical robot with equal dexterity and similar control. This can allow for more complex procedures to be completed that require various end effectors without the need to remove and insert new tools into the peritoneal cavity.

The surgeon can control the default end effector of the robot to perform its typical surgical task. Once the surgeon needs the use of the other end effector, the proximal joint can be rotated 180 degrees. Once this occurs, the surgeon is able to use the second end

effector with the same dexterity and workspace volume as the first end effector. A similar process allows for the robot to be reoriented to switch back to use of the first end effector. This process is shown in Figure 22.

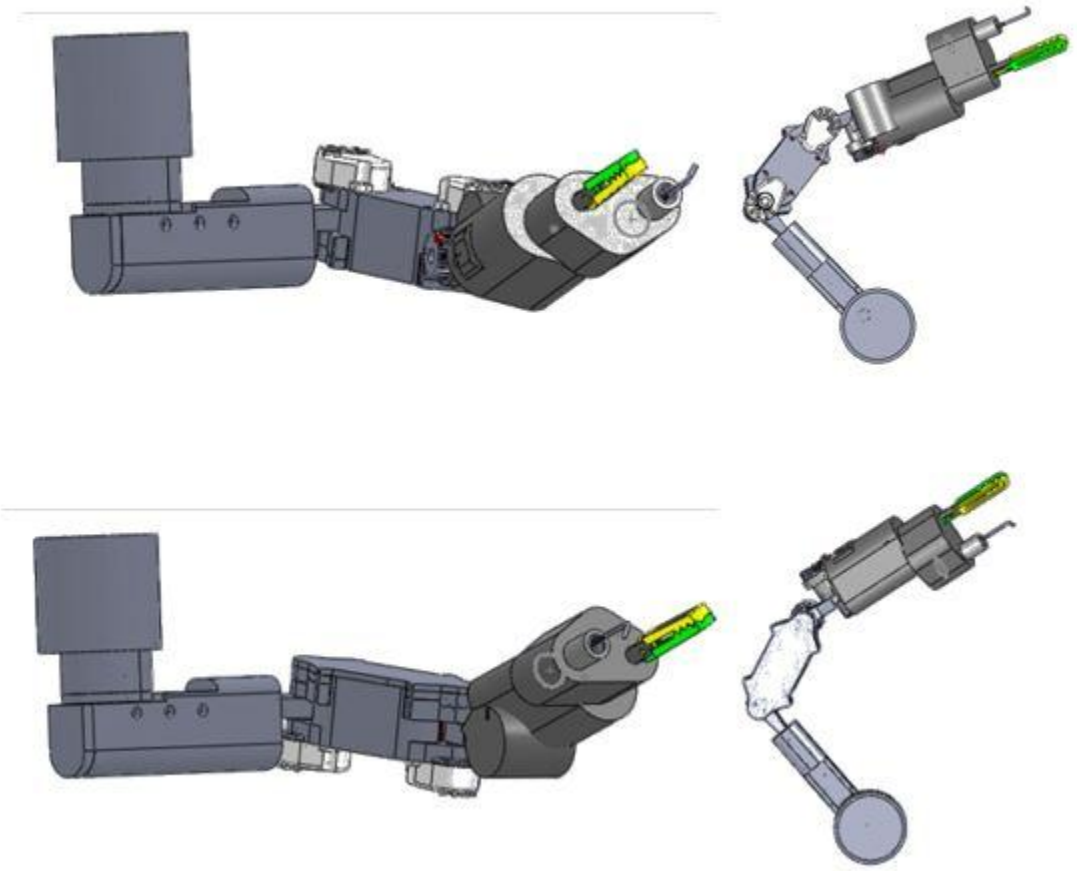

Figure 22: Side and top view of the dual end effector implementation on the left robotic arm

In the forearm, one motor will control the rotation of the end effector and one motor will control the open/close actuation. One end effector combination could include a monopolar cautery with a rotational degree of freedom plus a second end effector, such as a gripper or scissor, with a rotational degree of freedom and open/close actuation. One

motor will provide actuation for open/close of one end effector while a second motor will provide simultaneous rotation for both end effectors. For rotation, torque will be transmitted from the motor to the end effector shaft through spur gears. The open/close actuation will be accomplished through a lead screw mechanism. These mechanisms are a combination of both methods used for the individual cautery and grasper forearm as presented in the previous section.

Other options with various end effector combination could include two end effectors, both capable of both rotation and open/close actuation. In this configuration, one motor will provide simultaneous rotation of both end effectors, while a second motor will provide simultaneous open/close actuation for both end effectors. Other possibilities could include a linkage between the two end effectors to allow for both end effectors to be actuated between open and closed positions. Possible end effectors could include various graspers, needle drivers, scissors, cautery, Ligasure, or knife options.

The kinematics of each robotic arm allow for nearly identical dexterity and workspace volume for both end effectors. This is due to the placement of joints allowing for similar range of motion in both directions, as shown in Figure 23. When not in transition between end effectors, the proximal joint have full mobility. Because of this, when the arm is rotated into the orientation for use of the second end effector, the kinematics of the robot stay relatively the same, allowing for each of the two end effectors to be controlled by the surgeon in a similar way. Future work can lead to increased range of motion and larger dexterous workspace through rearrangement of components and joints.

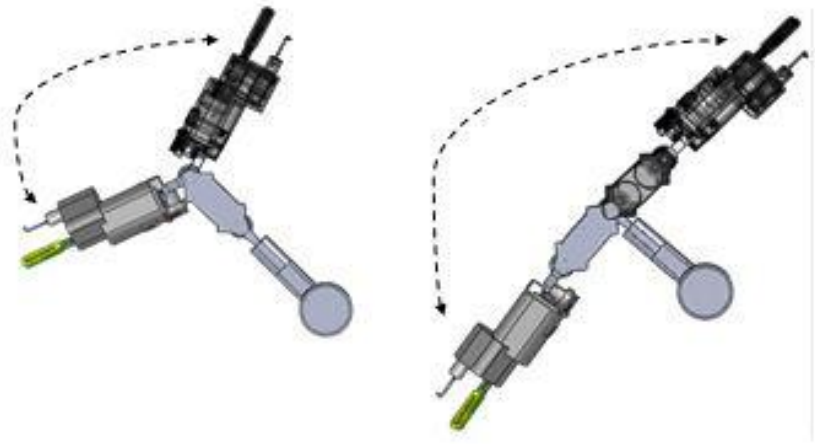

Figure 23: Bidirectional capability of joints

Volume of material at the end effector side of the forearm has been minimized to allow for better visualization of the end effector in use. The cross sectional area of the dual tip end effector forearm is compact enough to fit through a single incision for use in Laparoendoscopic Single-Site Surgery (LESS).

To better accommodate the use of both end effectors, a method to retract one of the end effectors will be used, as shown in Figure 24. When the first end effector is in use, the second end effector will be partially retracted to avoid unwanted collisions with tissue of the second end effector. Once the second end effector is needed for use, it will extend out from the forearm while the arm reorients itself using the reorientation protocol. The second end effector will be extended beyond the first end effector to allow use without the first end effector colliding with tissue. Once the use of the second end effector is completed, it will retract back into the forearm as the arm reorients itself for use of the first end effector.

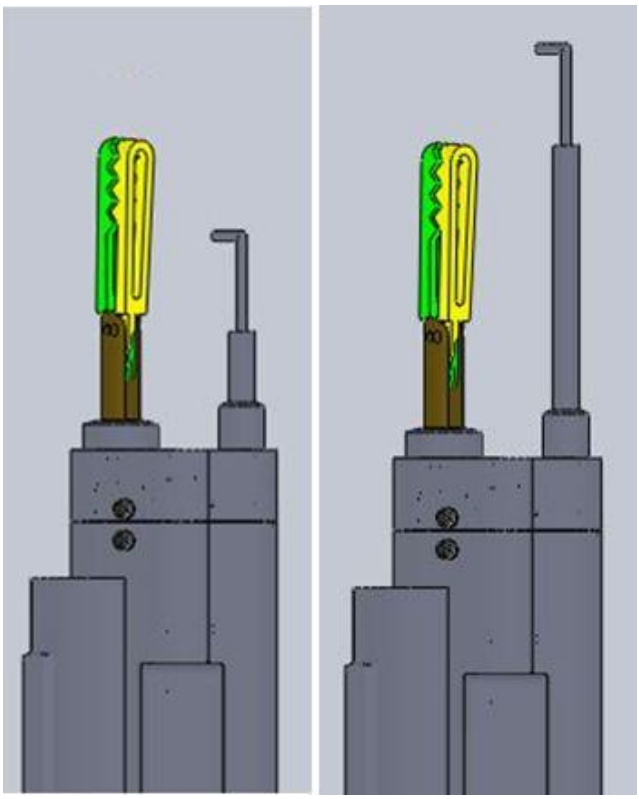

Figure 24: Retracted and extended cautery end effector

This mechanism is better demonstrated in Figures 25 and 26. The rotational spur gear causes the rotation of the long cautery shaft. The long cautery shaft, which is supported by two ball bearings, is mated with the retractable cautery shaft. As this rotation occurs, the externally threaded retractable cautery shaft is threaded into the threaded energizing ring, advancing the retractable cautery shaft through a combined rotation and translation. The threaded energizing ring is rigidly constrained to the forearm motor housing. The power supply for the cautery is attached to the threaded energizing ring via a wire to conduct the energy required for use of the cautery to the hook cautery tip. A rod supports the retractable cautery shaft to assist in maintaining proper mating between the retractable cautery shaft and the long cautery shaft.

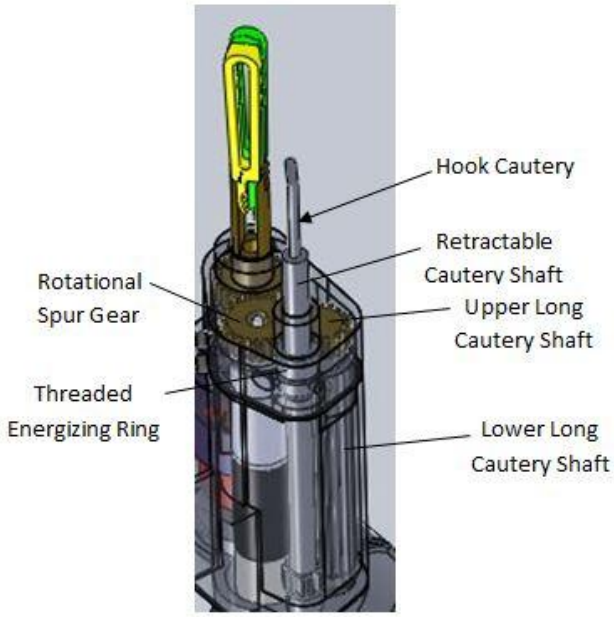

Figure 25: Transparent retractable cautery view

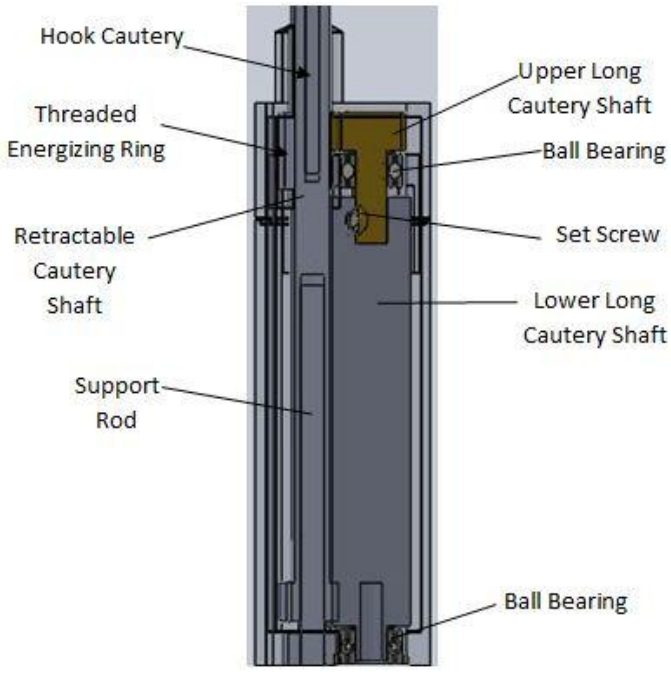

Figure 26: Retractable cautery mechanism cross sectional view

Chapter 6: *In Vivo* Results

This six DOF miniature *in vivo* surgical robot has been used in three non-survival surgical procedures in a live porcine model at the University of Nebraska Medical Center. All cases were approved by the institutional review board. During these procedures, the robot was controlled by a surgeon at the remote surgical user interface located in the surgery room away from the porcine model.

Section 6.1: Insertion and Attachment

The surgical robot prototype was successfully inserted into the peritoneal cavity during three separate procedures in live porcine models. Each arm was inserted individually before being coupled together using a central assembly rod. The incision in the peritoneum in each of these procedures was approximately 60 mm to provide initial proof of concept. In the future, this incision size will be decreased. The steps of this process are shown in Figure 27. After the incision was made, the insertion procedure took an average of five minutes for insertion and insufflation.

Figure 27: Insertion of surgical robot into porcine peritoneal cavity

Section 6.2: Surgical Procedures

6.2.1: Cholecystectomy

The surgical robot was used to perform a single incision cholecystectomy, or gallbladder removal. Once the robot was inserted through a single incision into a live porcine model, a standard laparoscope was inserted through the Gel Port in such a way that it utilized the original single incision by going through a cutout in the assembly rod. This cutout was designed to allow for four degree of freedom motion of the laparoscope inside the cavity to provide visual feedback to the surgeon. Once the robot was grossly positioned with use of the protruding assembly rod, the liver was retracted utilizing a 5 mm laparoscopic grasper through an additional laparoscopic port. In future applications, this retraction could be performed with a scarless 3 mm retractor to maintain single incision classification.

Once the robot was in position and the liver was retracted, the surgeon took his place at the remote surgical user interface made up of the master controller, monitor, and triple-action footpedal. Additionally, a standard cautery foot pedal was used to control the cut and coagulation abilities of cautery.

The first step of the cholecystectomy was the separation of the cystic duct from the liver bed. After this step was complete, the cystic duct was dissected. Finally, the gallbladder tissue was dissected from the liver bed until the gallbladder was completely

Figure 28: *In vivo* cholecystectomy images

detached. Once the procedure was completed, the robot and gallbladder were removed from the body through the single incision. Images from this procedure, as seen through the laparoscopic view displayed on the monitor of the surgical user interface, are shown in Figure 28.

6.2.2: Colon Resection

This robot was also used to perform a colon resection on the live porcine model. After the robot was inserted and the peritoneal cavity was insufflated, the left grasper end effector was used to grasp portions of the colon and provide tension in the mesentery when pulled. Once tension in the mesentery was achieved, the right hook cautery end effector was utilized to dissect the mesentery in order to mobilize the colon. Images, as seen by the surgeon in real time via the monitor, are shown in Figure 29.

Figure 29: *In vivo* colectomy images

6.2.3: Small Bowel Resection

Due to the differences in the anatomy of the colon of a human and porcine model, the small bowel in a porcine model is used to simulate the right colon found in a human. For this procedure, the left arm's grasper end effector was used to grasp a portion of small bowel and pull the tissue to create tension in the mesentery. The right arm's hook cautery end effector was then used to dissect the mesentery in order to mobilize the small bowel, as would be required for human colon resection. Laparoscopic images from this procedure as viewed by the surgeon through the monitor at the remote surgical user interface are shown in Figure 30.

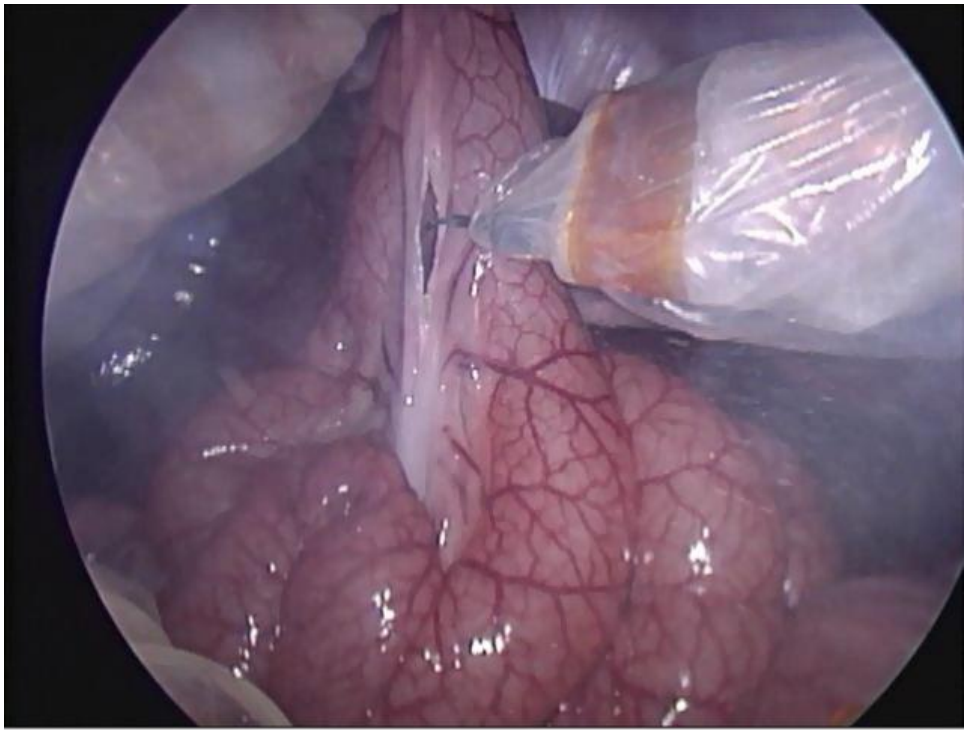

Figure 30: *In vivo* small bowel resection image

Chapter 7: Conclusion

This thesis presented a six DOF miniature *in vivo* surgical robot designed for LESS. The robot has the capabilities of being inserted through a single incision and performing surgical tasks over a large, multiple quadrant workspace. Once inserted inside the peritoneal cavity, the robot is controlled via a remote surgical user interface. Feasibility has been demonstrated in multiple LESS procedures, including a cholecystectomy, colon resection, and small bowel resection.

Future work will focus on implementing the retractable cautery dual end effector forearm and performing procedures in multiple quadrants without external repositioning. To complete this, a new onboard camera will likely be needed capable of providing visual feedback in multiple quadrants. A control system will also need to be implemented to provide the surgeon with acceptable switching of dual end effectors. Additional work will also be performed to improve robustness and reliability of the platform to the level required for FDA approval.

.

References

[1] Delany, C., Neary, P., Heriot, A., Senagore, A., 2007. Operative Techniques in Laparoscopic Colorectal Surgery. Lippincott Williams & Wilkins, Philadelphia, PA.

[2] Horgan, S., and Vanuno, D. 2001. Robots in laparoscopic surgery. Journal of Laparoendoscopic & Advanced Surgical Techniques. Part A, 11(6):415-419.

[3] Mack, M. J., et al. 1992. Present role of thoracoscopy in the diagnosis and treatment of diseases of the chest. The Annals of Thoracic Surgery, 54(3):403-8.

[4] Kim, V. B., et al. 2002. Early experience with telemanipulative robot-assisted laparoscopic cholecystectomy using da vinci. Surgical Laparoscopy, Endoscopy & Percutaneous Techniques, 12(1):33-40.

[5] Ahmed K., Wang T. T., Patel V. M., Nagpal K., Clark J., Ali M., Deeba S., Ashrafian H., Darzi A., Athanasiou T., Paraskeva P. 2011. The role of single-incision laparoscopic surgery in abdominal and pelvic surgery: a systematic review. Surgical Endoscopy. 25(2):378-96.

[6] Fader, A. N., Levinson, K. L., Gunderson, C. C., Winder, A. D., Escobar, P. F. 2011. Laparendoscopic single-site surgery in gynaecology: A new frontier in minimally invasive surgery. Journal of Minimal Access Surgery. 7(1): 71-77.

[7] Satava, R. M. 2002. Surgical robotics: The early chronicles: A personal historical perspective. Surgical Laparoscopy, Endoscopy & Percutaneous Techniques, 12(1):6-16.

[8] Ballantyne, G.H., 2002. Robotic Surgery, Telerobotic Surgery, Telepresence, and Telementoring. *Surgical Endoscopy*, 16, pp. 1389-1402.

[9] Corcione, F., Esposito, C., Cuccurullo, D., Settembre, A., Miranda, N., Amato, F., Pirozzi, F., Caiazzo, P., 2005. Advantages and Limits of Robot-Assisted Laparoscopic Surgery. *Surgical Endoscopy*, 19, pp. 117-119.

[10] Moorthy, K., Munz, Y. Dosis, A., Hernandez, J., Martin, S., Bello, F., Rockall, T., Darzi, A., 2004. Dexterity Enhancement with Robotic Surgery. *Surgical Endoscopy*, 18, pp. 790-795.

[11] Lum, M., Friedman, D., Sankaranarayanan, G., King, H., Fodero, K., Leuschke, R., Hannaford, B., Rosen, J., Sinanan, M., 2009. The RAVEN – Design and Validation of a Telesurgery system. *International Journal of Robotics Research*, 28, Online.

[12] Nelson, C., Zhang, X., Shah, B.C., Goed, M.R., Oleynikov, D., 2010. Multipurpose Surgical Robot as a Laparoscope Assistant. *Surgical Endoscopy*, 24(7): 1528-1532.

[13] Ostrowitz, M., Eschete, D., Zemon, H., DeNoto, G., 2009. Robotic-Assisted Single-Incision Right Colectomy: Early Experience. *International Journal of Medical Robotics and Computer Assisted Surgery*, 5, 465-470.

[14] Piccigallo, M., Scarfogliero, U.,Quaglia, C., Petroni, G.,Valdastri, P., Menciassi, A., Dario, P. 2010. Design of a Novel Bimanual Robotic System for Single-Port Laparoscopy. IEEE/ASME Transcations on Mechatronics. 15(6): 871-878.

[15] Rentschler M, Hadzialic A, Dumpert J, Platt S, Farritor S, Oleynikov D. *In vivo* robots for laparoscopic surgery. In: Studies in Health Technology and Informatics –Medicine Meets Virtual Reality IOS Press, Newport Beach CA 2004. vol. 98, pp.316-322.

[16] Oleynikov D, Rentschler M, Hadzialic A, Dumpert J, Platt S, Farritor S. Miniature robots can assist in laparoscopic cholecystectomy. Surgical Endoscopy 2005; 9(4):473–476.

[17] Hawks J. "Improved mobile wireless in vivo surgical robots: Modular design, experimental results, and analysis" (January 1, 2010). *ETD collection for University of Nebraska - Lincoln.* Paper AAI3432048.

[18] McCormick R, Wortman T, Strabala K, Frederick T, Oleynikov D, Farritor S, 2011. Kinematic and Workspace Comparison of Four and Five Degree of Freedom Miniature In Vivo Surgical Robot. In Proceedings of the 2011Design of Medical Devices Conference, Minneapolis, MN.

[19] Cha A, 2010. The top 10 deadliest cancers – and why there's no cure. MSNBC. Retrieved June 22, 2010, from MSNBC.com.

[20] Rosen J, Solazzo M, Hannaford B, Sinanan M, Objective Evaluation of Laparoscopic Surgical Skills Using Hidden Markov Models Based on Haptic Information and Tool/Tissue Interactions. American College of Surgeons Annual Meeting - Washington State Chapter, Lake Chelan, June 2000.

[21] Rosen J, Brown JD, Barreca M, Chang L, Sinanan M, Hannaford B. The BlueDRAGON – A System of Measuring the Kinematics and Dynamics of Minimally Invasive Surgical Instruments In-Vivo. In: Proc. IEEE International Conference on Robotics and Automation, Washington, DC, 2002, pp. 1876-1881.

[22] Asada H. *Robot Analysis and Control*. Chichester: John Wiley & Sons, 1986.

[23] Craig J. *Introduction to Robotics: Mechanics and Control*. Reading, MA: Addison-Wesley, 1989. Print.

Appendix A. Kinematic Design Supporting Material

MATLAB Code to Determine Forward Kinematics and Jacobian:

```
clc
clear

syms theta1 theta2 theta3 theta4 theta5 theta6 theta7;
%syms sin(theta1) sin(theta2) sin(theta3) sin(theta4) sin(theta5)
sin(theta6) sin(theta7);
syms Lua Lla Lfa real;

P5e = [Lfa 0 0]';

T01 = [cos(theta1) -sin(theta1) 0 0; sin(theta1) cos(theta1) 0 0; 0 0 1
0; 0 0 0 1];

T12 = [cos(theta2) -sin(theta2) 0 0; 0 0 1 -Lua; -sin(theta2) -
cos(theta2) 0 0; 0 0 0 1];

T23 = [cos(theta3) -sin(theta3) 0 0; 0 0 -1 0; sin(theta3) cos(theta3)
0 0; 0 0 0 1];

T34 = [cos(theta4) -sin(theta4) 0 Lla; sin(theta4) cos(theta4) 0 0; 0 0
1 0; 0 0 0 1];

T45 = [cos(theta5) -sin(theta5) 0 0; 0 0 -1 0; sin(theta5) cos(theta5)
0 0; 0 0 0 1];

T56 = [0 0 1 0; -sin(theta6) -cos(theta6) 0 0; cos(theta6) -sin(theta6)
0 0; 0 0 0 1];

T02 = T01*T12;
T03 = T02*T23;
T04 = T03*T34;
T05 = T04*T45;
T06 = T05*T56;
T46 = T45*T56;
T36 = T34*T46;
T26 = T23*T36;
T16 = T12*T26;

T35 = T34*T45;
T25 = T23*T35;
T15 = T12*T25;

P0e = T05*[P5e;1]

bbar = [0 0 1]';


% i = 1
b0 = bbar;
```

```
JA1 = b0;

r0 = T05*[P5e; 1];
r0 = r0(1:3);

JL1 = cross(b0, r0);


% i = 2
b1 = T01(1:3,1:3)*bbar;
JA2 = b1;

r1 = T15*[P5e; 1];
r1 = r1(1:3);

JL2 = cross(b1, r1);


% i = 3
b2 = T02(1:3,1:3)*bbar;
JA3 = b2;

r2 = T25*[P5e; 1];
r2 = r2(1:3);

JL3 = cross(b2, r2);


% i = 4
b3 = T03(1:3,1:3)*bbar;
JA4 = b3;

r3 = T35*[P5e; 1];
r3 = r3(1:3);

JL4 = cross(b3, r3);


% i = 5
b4 = T04(1:3,1:3)*bbar;
JA5 = b4;

r4 = T45*[P5e; 1];
r4 = r4(1:3);

JL5 = cross(b4, r4);


% i = 6
b5 = T05(1:3,1:3)*bbar;
JA6 = b5;

r5 = P5e;
```

```
JL6 = cross(b5, r5)

JA = [JA1 JA2 JA3 JA4 JA5 JA6];
JL = [JL1 JL2 JL3 JL4 JL5 JL6];

J = [JL; JA]

JT = transpose(J);

for k = 1:6
    for i = 1:6
        J(i,k);
    end
end
```

MATLAB Code for Force and Velocity Plots:

```
clear all
close all
clc


%dlmwrite('manipdata.dat', ['X', 'Y', 'Z', 'w'])

%% Joint Parameters
theta1min=-89*pi/180;
theta1max=89*pi/180;
theta1step=10*pi/180;

theta2min=-90*pi/180;
theta2max=90*pi/180;
theta2step=10*pi/180;

theta3min=-90*pi/180;
theta3max=90*pi/180;
theta3step=10*pi/180;

theta4min=-90*pi/180;
theta4max=90*pi/180;
theta4step=10*pi/180;

theta5min=-90*pi/180;
theta5max=90*pi/180;
theta5step=10*pi/180;

theta6min=-90*pi/180;
theta6max=90*pi/180;
theta6step=10*pi/180;


%%Robot Specs (mm)
Lua = 65.4;
Lla = 41;
Lfa = 63.7;
%% Motor Specs mNm
T(1) = 1220.61;
T(2) = 490;
T(3) = 264.17;
T(4) = 264.17;
T(5) = 264.17;
T(6) = 264.17;

wm(1) = .97; %rad/s
wm(2) = 1.69;
wm(3) = 1.11;
wm(4) = 1.11;
wm(5) = 1.11;
wm(6) = 6.46;

halfxsection = .25;
```

```
l=0; m=0; n=0; p=0; q=0; Xn=0; nn=0;
for theta1= theta1min:theta1step:theta1max
    l=l+1;
    m=0;
    for theta2= theta2min:theta2step:theta2max
        m=m+1;
        n=0;
        for theta3= theta3min:theta3step:theta3max
            n=n+1;
            p=0;
                for theta4= theta4min:theta4step:theta4max
                 p=p+1;
                 q=0;
                    for theta5= theta5min:theta5step:theta5max
                 q=q+1;



            %% Forward Kinematics
            XF(n,p,q) =  Lua*sin(theta1) - Lla*(sin(theta1)*sin(theta3)
- cos(theta1)*cos(theta2)*cos(theta3)) -
Lfa*(cos(theta5)*(cos(theta4)*(sin(theta1)*sin(theta3) -
cos(theta1)*cos(theta2)*cos(theta3)) +
sin(theta4)*(cos(theta3)*sin(theta1) +
cos(theta1)*cos(theta2)*sin(theta3))) -
cos(theta1)*sin(theta2)*sin(theta5));
            YF(n,p,q) =   Lla*(cos(theta1)*sin(theta3) +
cos(theta2)*cos(theta3)*sin(theta1)) +
Lfa*(cos(theta5)*(cos(theta4)*(cos(theta1)*sin(theta3) +
cos(theta2)*cos(theta3)*sin(theta1)) +
sin(theta4)*(cos(theta1)*cos(theta3) -
cos(theta2)*sin(theta1)*sin(theta3))) +
sin(theta1)*sin(theta2)*sin(theta5)) - Lua*cos(theta1);
            ZF(n,p,q) =  Lfa*(cos(theta2)*sin(theta5) +
cos(theta5)*(sin(theta2)*sin(theta3)*sin(theta4) -
cos(theta3)*cos(theta4)*sin(theta2))) - Lla*cos(theta3)*sin(theta2);



 %% Jacobian Calculation Frame Zero



            J(1,1) = Lua*cos(theta1) -
Lfa*(cos(theta5)*(cos(theta4)*(cos(theta1)*sin(theta3) +
cos(theta2)*cos(theta3)*sin(theta1)) +
sin(theta4)*(cos(theta1)*cos(theta3) -
cos(theta2)*sin(theta1)*sin(theta3))) +
sin(theta1)*sin(theta2)*sin(theta5)) - Lla*(cos(theta1)*sin(theta3) +
cos(theta2)*cos(theta3)*sin(theta1));
            J(2,1) = Lua*sin(theta1) - Lla*(sin(theta1)*sin(theta3) -
cos(theta1)*cos(theta2)*cos(theta3)) -
Lfa*(cos(theta5)*(cos(theta4)*(sin(theta1)*sin(theta3) -
cos(theta1)*cos(theta2)*cos(theta3)) +
```

```
sin(theta4)*(cos(theta3)*sin(theta1) +
cos(theta1)*cos(theta2)*sin(theta3))) -
cos(theta1)*sin(theta2)*sin(theta5));
            J(3,1) = 0;
            J(4,1) = 0;
            J(5,1) = 0;
            J(6,1) = 1;
            J(1,2) = Lua - Lla*sin(theta3) -
Lfa*(cos(theta3)*cos(theta5)*sin(theta4) +
cos(theta4)*cos(theta5)*sin(theta3));
            J(2,2) = Lfa*(sin(theta2)*sin(theta5) -
cos(theta2)*(cos(theta5)*sin(theta3)*sin(theta4) -
cos(theta3)*cos(theta4)*cos(theta5))) + Lla*cos(theta2)*cos(theta3);
            J(3,2) = 0;
            J(4,2) = 0;
            J(5,2) = 0;
            J(6,2) = 1;
            J(1,3) = cos(theta1)*(Lla*sin(theta3) +
Lfa*(cos(theta3)*cos(theta5)*sin(theta4) +
cos(theta4)*cos(theta5)*sin(theta3)));
            J(2,3) = sin(theta1)*(Lla*sin(theta3) +
Lfa*(cos(theta3)*cos(theta5)*sin(theta4) +
cos(theta4)*cos(theta5)*sin(theta3)));
            J(3,3) = Lfa*sin(theta1)*sin(theta5) -
cos(theta1)*(Lla*cos(theta3) - Lfa*(cos(theta5)*sin(theta3)*sin(theta4)
- cos(theta3)*cos(theta4)*cos(theta5)));
            J(4,3) = -sin(theta1);
            J(5,3) = cos(theta1);
            J(6,3) = 0;
            J(1,4) = Lfa*sin(theta1)*sin(theta2)*sin(theta5) -
Lfa*cos(theta2)*cos(theta5)*sin(theta4);
            J(2,4) = cos(theta2)*(Lla + Lfa*cos(theta4)*cos(theta5)) -
Lfa*cos(theta1)*sin(theta2)*sin(theta5);
            J(3,4) =
Lfa*cos(theta1)*cos(theta5)*sin(theta2)*sin(theta4) -
sin(theta1)*sin(theta2)*(Lla + Lfa*cos(theta4)*cos(theta5));
            J(4,4) =  cos(theta1)*sin(theta2);
            J(5,4) = sin(theta1)*sin(theta2);
            J(6,4) = cos(theta2);
            J(1,5) = Lfa*sin(theta1)*sin(theta2)*sin(theta5);
            J(2,5) = Lfa*cos(theta2)*cos(theta5) -
Lfa*cos(theta1)*sin(theta2)*sin(theta5);
            J(3,5) = -Lfa*cos(theta5)*sin(theta1)*sin(theta2);
            J(4,5) = cos(theta1)*sin(theta2);
            J(5,5) = sin(theta1)*sin(theta2);
            J(6,5) = cos(theta2);
            J(1,6) = 0;
            J(2,6) = -Lfa*(cos(theta3)*sin(theta2)*sin(theta4) +
cos(theta4)*sin(theta2)*sin(theta3));
            J(3,6) = Lfa*(cos(theta4)*(cos(theta1)*cos(theta3) -
cos(theta2)*sin(theta1)*sin(theta3)) -
sin(theta4)*(cos(theta1)*sin(theta3) +
cos(theta2)*cos(theta3)*sin(theta1)));
            J(4,6) = cos(theta4)*(cos(theta3)*sin(theta1) +
cos(theta1)*cos(theta2)*sin(theta3)) -
sin(theta4)*(sin(theta1)*sin(theta3) -
cos(theta1)*cos(theta2)*cos(theta3));
```

```
            J(5,6) = sin(theta4)*(cos(theta1)*sin(theta3) +
cos(theta2)*cos(theta3)*sin(theta1)) -
cos(theta4)*(cos(theta1)*cos(theta3) -
cos(theta2)*sin(theta1)*sin(theta3));
            J(6,6) = - cos(theta3)*sin(theta2)*sin(theta4) -
cos(theta4)*sin(theta2)*sin(theta3);

  %% Manipulability Measure
            w(n,p,q) = sqrt(abs(det(J*transpose(J))))/9775100000000000;

 %% Force and Velocity
          F = ((J')^-1)*T'; % mNm/mm
            F =inv(transpose(J))*T';
            Fx(n,p,q)=abs(F(1));
            Fy(n,p,q)=abs(F(2));
            Fz(n,p,q)=abs(F(3));

            V = J*wm';
            Vx(n,p,q)=abs(V(1));
            Vy(n,p,q)=abs(V(2));
            Vz(n,p,q)=abs(V(3));

            % Find minimum values
            %Fm(n,p,q),Fp(n,p,q) = min(abs(F));

            %Vm(l,m,n),Vp(l,m,n) = min(abs(V));

                    end
                end

        end



    %% Plot Workspace Mesh
% figure(1)
% hold on
% s = 30;
% for a=1:size(YF,1)
%    for b=1:size(YF,2)
%        for c=1:size(YF,3)
%            if abs(YF(a,b,c)) < 0.1
%                if (XF(a,b,c)) > 0
%                    if w(a,b,c) > .1
%
scatter3(XF(a,b,c),YF(a,b,c),ZF(a,b,c),s,w(a,b,c),'filled');
%                    %csvwrite('csv.dat',[XF(a,b,c), YF(a,b,c),
ZF(a,b,c), w(a,b,c)]);
%                    %dlmwrite('manipdata.xls',[XF(a,b,c), YF(a,b,c),
ZF(a,b,c), w(a,b,c)]);
%                    dlmwrite('manipdata.dat', [XF(a,b,c), ZF(a,b,c),
w(a,b,c)], '-append')
%                    end
%                end
%            end
```

```
%        end
%    end
% end

%%Force Plot


figure(1)
hold on
for a=1:size(YF,1)
   for b=1:size(YF,2)
       for c=1:size(YF,3)
           if abs(YF(a,b,c)) < halfxsection
               if(XF(a,b,c)) > 0
                   if (Fx(a,b,c)) > 5
                   scatter(XF(a,b,c),ZF(a,b,c), 'k', 'filled');
                   end
               end
           end
       end
   end
end

%% view(1);
view(0,90);
axis([0 170 -120 120])
%colorbar;
%colormap('default')
xlabel('X mm')
ylabel('Z mm')

title('Fx')

%%Force Plot
figure(2)
hold on
for a=1:size(YF,1)
   for b=1:size(YF,2)
       for c=1:size(YF,3)
           if abs(YF(a,b,c)) < halfxsection
               if(XF(a,b,c)) > 0
                   if (Fy(a,b,c)) > 5
                   scatter(XF(a,b,c),ZF(a,b,c), 'k', 'filled');
                   end
               end
           end
       end
   end
end


%% view(1);
view(0,90);
axis([0 170 -120 120])
%colorbar;
```

```
%colormap('default')
xlabel('X mm')
ylabel('Z mm')
title('Fy')

%%Force Plot
figure(3)
hold on
for a=1:size(YF,1)
   for b=1:size(YF,2)
       for c=1:size(YF,3)
           if abs(YF(a,b,c)) < halfxsection
               if(XF(a,b,c)) > 0
                   if (Fz(a,b,c)) > 20
                   scatter(XF(a,b,c),ZF(a,b,c), 'k', 'filled');
                   end
               end
           end
       end
   end
end

%% view(1);
view(0,90);
axis([0 170 -120 120])
%colorbar;
%colormap('default')
xlabel('X mm')
ylabel('Z mm')
title('Fz')

%%Velocity Plot
figure(4)
hold on
for a=1:size(YF,1)
   for b=1:size(YF,2)
       for c=1:size(YF,3)
           if abs(YF(a,b,c)) < halfxsection
               if(XF(a,b,c)) > 0
                   if (Vx(a,b,c)) > 72 %.072 m/s = 72 mm/s
                   scatter(XF(a,b,c),ZF(a,b,c), 'k', 'filled');
                   end
               end
           end
       end
   end
end


%% view(1);
view(0,90);
axis([0 170 -120 120])
%colorbar;
%colormap('default')
xlabel('X mm')
ylabel('Z mm')
```

```
title('Vx')

%%Velocity Plot
figure(5)
hold on
for a=1:size(YF,1)
   for b=1:size(YF,2)
       for c=1:size(YF,3)
           if abs(YF(a,b,c)) < halfxsection
               if(XF(a,b,c)) > 0
                   if (Vy(a,b,c)) > 72 %.072 m/s = 72 mm/s
                   scatter(XF(a,b,c),ZF(a,b,c), 'k', 'filled');
                   end
               end
           end
       end
   end
end


%% view(1);
view(0,90);
axis([0 170 -120 120])
%colorbar;
%colormap('default')
xlabel('X mm')
ylabel('Z mm')
title('Vy')

%%Velocity Plot
figure(6)
hold on
for a=1:size(YF,1)
   for b=1:size(YF,2)
       for c=1:size(YF,3)
           if abs(YF(a,b,c)) < halfxsection
               if(XF(a,b,c)) > 0
                   if (Vz(a,b,c)) > 72 %.072 m/s = 72 mm/s
                   scatter(XF(a,b,c),ZF(a,b,c), 'k', 'filled');
                   end
               end
           end
       end
   end
end

%% view(1);
view(0,90);
axis([0 170 -120 120])
%colorbar;
%colormap('default')
xlabel('X mm')
ylabel('Z mm')
title('Vz')
    end
```

```
    display(theta1)

end
```